An Analysis-by-Synthesis Approach to Sinusoidal Modeling Applied to Speech and Music Signal Processing

A THESIS

Presented to

The Academic Faculty

By

E. Bryan George

In Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy in Electrical Engineering

Georgia Institute of Technology

November, 1991

An Analysis-by-Synthesis Approach to Sinusoidal Modeling Applied to Speech and Music Signal Processing

Approved:

Mark J. T. Smith, Chairman

Ronald W. Schafer

Mark A. Richards

Date approved by Chairman 12/2/91

Dedicated to the memories of:

Eric Bryan George, Jr.

Eugene L. Cornett, Sr.

Acknowledgments

First and foremost, I would like to thank my advisor, Professor Mark J. T. Smith, for providing guidance throughout my tenure as a graduate student. His extensive knowledge of digital signal processing has kept me out of many blind research alleys, and his insightful comments on my various papers and this thesis have significantly enhanced the quality of my published work. Furthermore, his skill as a fencer was instrumental in my eventually leaving Georgia Tech.[1] Many thanks go to the other members of my reading committee, Profs. Ronald W. Schafer and Mark A. Richards for their input on the thesis, and to the other members of my defense committee, Profs. Douglas B. Williams and Alfred D. Andrew.

My appreciation and respect goes out to many past and present colleagues at Georgia Tech, particularly Beth Carlson, Dave Mazel, Brian Evans, Sam Liu, Kate Maloney, Dave Pepper, Janet Rutledge and Pete Wung. I would like to acknowledge the staff of the Georgia Tech Office of Graduate Studies, most notably Dr. Helen Grenga and Ms. Glenna Thomas, whose assistance and suggestions in my final quarters enabled me to finish my thesis a thousand miles away from school and several thousand miles away from my advisor.

I wish to pay tribute to Mr. Mike Harben, who originally observed the possible musical applications of my work, and to the remarkably large group of musically talented graduate students in the DSP group who assisted me with my research efforts in music synthesis, especially Larry Heck, Eric Hyche and Lonnie Harvel. On a final musical note, a special salute goes to the DSP bluegrass group: Tom Barnwell, Sam Smith, and Craig Richardson – we were never all that good, but we were *Close Enough for Bluegrass*.

The patience and devotion of my wife, Robin, is beyond my ability to acknowledge. The best I can do is simply to say that she never doubted the wisdom of my decision to attend graduate school or underrated my goal to become a research engineer. In addition, we owe a great deal of love and thanks to our families and to the many friends we have made in Atlanta, especially Fredy Newton, Mary Ann French, Mike and Fran Harben, John and Cat Helms, Lois Hertz and Jim Davis. We've been through both triumphs and tragedies together, and your support has made the dream possible – we won't forget it.

[1] Just wanted to see if you were paying attention.

Finally, I wish to acknowledge the profound influence of Eugene Cornett, my high school vocational instructor, on my career. Not only did his teaching initially spark my interest in an engineering career, but the ability I gained in his classes, to see electrical engineering from the perspective of a technician as well as a theoretician, is the greatest gift any researcher can have.

This work was supported by the Jet Propulsion Laboratory, Pasadena CA, and by the National Science Foundation under contract DCI-8611372.

Contents

List of Tables

List of Figures

Summary

Sinusoidal modeling has been successfully applied to a broad range of signal processing problems, and offers advantages over linear predictive modeling and the short-time Fourier transform for the analysis, synthesis and modification of speech and music signals. However, the most popular system used in sinusoidal modeling, the *Sine-wave Transform System*, relies on an analysis procedure which estimates parameters by identifying the peaks of an interpolated discrete Fourier transform, and uses a synthesis model which is based on interpolating these estimated parameters over time. The objectives of this thesis are to consider whether a different analysis technique can more accurately determine sinusoidal model parameters (particularly in transitory regions), to explore whether a sinusoidal model formulation that correlates well with the proposed analysis procedure is useful in speech modification and music synthesis applications, and to determine if the proposed analysis and synthesis techniques may be implemented in a computationally efficient form.

The research presented here improves the usefulness of sinusoidal models in digital signal processing by investigating the use of an analysis-by-synthesis procedure to determine the parameters of an overlap-add sinusoidal model formulation, and by developing this analysis/synthesis system for use in a variety of digital signal processing applications. Specifically, the contributions of this work include (1) introduction of an analysis/synthesis system which combines an accurate and robust analysis-by-synthesis procedure with an overlap-add sinusoidal model formulation, (2) derivation of a refined overlap-add sinusoidal model formulation capable of modifying speech and music signals without artifacts, (3) application of the developed system to the problems of speech modification and music synthesis, and (4) development of tech-

niques to reduce the computational load required in the analysis/synthesis system. The developed system was tested on a set of test utterances and musical tones in software simulations, and its performance was evaluated relative to that of the Sine-wave Transform System using both objective and subjective measures.

CHAPTER 1

Introduction and Background

1.1 Problem Statement

One of the most important problems in digital signal processing is that of representing one-dimensional discrete-time sequences using *parametric signal models*. Broadly defined, a parametric signal model is a fixed mathematical construct which represents signals in terms of a set of variables or "parameters." In digital signal processing, the goal of signal modeling is to design a representation whose parameters can be more effectively processed than the original signal for a given purpose. This thesis will focus on the definition and implementation of a parametric model appropriate for processing audio signals.

The need for parametric signal models can be seen by considering speech as an example. Digital speech processing has long been an active area of both theoretical research and practical application, particularly in the telecommunications field. One of the more interesting speech processing problems is low bit-rate speech coding, where the objective is to transmit high-quality speech over digital communication channels at the lowest possible transmission rate.

While success has been achieved by manipulating and coding speech waveforms directly, there is much redundant information in speech for which such techniques cannot account. For this reason, parametric signal models which represent aspects of the speech production and human auditory processes have been widely used in high-quality speech coders operating below 8000 bits/sec; the most well-known cod-

ing technique using a parametric model is *linear predictive coding* (LPC) [1]. The advantage of linear predictive models for coding applications is that their parameters relate directly to important (and easily coded) parameters of speech production, such as pitch, voicing state, and configuration of the vocal tract.

Another important digital speech processing problem is that of *speech modification*. Generally speaking, speech modification refers to the process of changing some perceptual property of a given speech signal without affecting other properties or speech quality. Speech modification is used in a variety of applications related to speech communication, such as speech enhancement, bandwidth reduction, aids to the hearing impaired, and man-machine communication. Among the many types of speech information which may be modified are pitch, rate of articulation, message content and speaker identity.

Unfortunately, this embedded information is not easily separable when dealing directly with time-domain signals, thus the simple manipulation of speech signals in order to alter one type of information often produces undesirable changes in other types of information as well. For instance, modification of speech properties such as articulation rate or fundamental frequency may be performed by altering the playback sampling rate of speech.[1] While this approach effectively changes articulation rate or fundamental frequency, it implies changing the two simultaneously, which is often undesirable. Furthermore, changes in the short-time speech spectrum brought about by altering playback rate degrade intelligibility and make speaker identification difficult.

As mentioned previously, the parameters of signal models can often be related directly to speech production. Therefore, parametric signal models provide an effective means of separating and organizing the information contained in speech signals in ways not possible using time-domain signals directly. For speech modification applications, this organization provides the ability to independently access and con-

[1]The digital equivalent of changing the speed of a record player.

trol speech properties, making parametric signal models extremely effective tools for speech processing. Two major objectives of this thesis will be the formulation of a parametric signal model for speech and associated techniques for determining model parameters given an input speech waveform, and on the application of this model to the areas of speech analysis/synthesis and speech modification.

Digital music synthesis is another application area which benefits greatly from advances in signal modeling techniques. For instance, both linear prediction models and the *digital phase vocoder* [2, 3] have been successfully applied to music synthesis and music signal processing applications. To date, however, there has existed a significant tradeoff between the quality of synthetic music and the simplicity of synthesis. While music signal processing has traditionally been a less popular topic of research than speech processing, the rapidly developing electronic musical instrument market, as well as demands in multimedia applications for computationally efficient, high-quality music synthesis and signal processing techniques are steadily increasing the importance of research in this area. Therefore, this thesis will also focus on the development of a system appropriate for the analysis, synthesis and modification of musical tones, based on the same system developed for speech signals.

1.2 Research Approach

The thesis research began with an exploration of the techniques presented by McAulay and Quatieri for modeling speech signals as a sum of sinusoidal components. Their sinusoidal model formulation, referred to as the *Sine-wave Transform System* (STS) [4], has proven to be useful in a wide range of speech processing applications [5, 6, 7, 8]. In addition, both the analysis and synthesis techniques used in the STS are well-justified and reasonable, given assumptions that are commonly made concerning speech signals. However, a thorough investigation and analysis of the Sine-wave Transform System revealed that for segments of speech in which these assumptions are not

valid. distortions can be introduced.

Two factors are responsible for distortions observed in the STS. The first arises due to the *peak-picking* analysis procedure used in the STS; in this procedure, frequency locations of significant spectral peaks are assumed to correspond to optimal component frequencies, and the amplitude and phase of spectral values at these frequencies are assumed to optimally represent the speech signal over a short time frame. While these assumptions may be somewhat justified for the case of steady-state speech, they result in suboptimal performance due to frequency-domain interference effects caused by windowing. Furthermore, the performance of peak-picking is sensitive to the validity of the assumed steady-state condition.

The second factor is the sinusoidal model used for speech synthesis. In the Sine-wave Transform System, analyzed frequency parameters from adjacent frames are matched using a "nearest-neighbor" algorithm; given matched pairs of frequencies, amplitude and phase parameters are interpolated along the resulting *frequency tracks*, yielding piecewise-continuous parameters. While the resulting functional form makes speech modification possible in the STS, the parameter set produced represents an uncontrolled departure from the theoretical basis of peak-picking analysis and is completely consistent with the analysis technique only in the case of steady-state speech. Furthermore, the nonlinear form of the model makes it very difficult to understand from a theoretical standpoint, making algorithm analysis and improvement difficult.

The next research step was to review recent advances in low bit-rate speech coding techniques. One of the most interesting approaches to LPC excitation analysis encountered is referred to as ***analysis-by-synthesis***. This technique, first introduced in the context of speech coding by Atal and Remde [9], has achieved dramatic performance gains in several linear predictive vocoders [10, 11, 12]. Much of the success of analysis-by-synthesis can be attributed to several factors: First, analysis-by-synthesis is a tractable approach to highly nonlinear or underconstrained optimization problems which are often encountered in signal modeling. Second, analysis-by-synthesis

is well-defined and easily understood in terms of vector space theory, making design analysis and improvement feasible. Third, analysis-by-synthesis allows properties of aural perception to be accounted for in the modeling of audio signals.

After examining various sinusoidal model formulations, it was quickly recognized that an overlap-add sinusoidal model might be viewed as an approximation to a signal vector, and that model parameters for this formulation could be determined using nonlinear approximation techniques. Since analysis-by-synthesis is particularly well-suited for nonlinear vector space approximation, it was decided to investigate the effects of applying analysis-by-synthesis to an overlap-add sinusoidal model for audio signal processing applications.

Preliminary experimentation with this approach yielded very positive results; however, the high computational requirements of analysis-by-synthesis seriously limited its utility for practical applications. Further research solved this problem by casting analysis-by-synthesis in the frequency domain and making use of the *Fast Fourier Transform* (FFT) algorithm. In addition, it was found that overlap-add synthesis could be performed using the *inverse FFT* (IFFT) algorithm, making audio analysis/synthesis using the modeling system practicable.

Given that the sinusoidal modeling system was to be applied primarily to the modification of audio signals, the next major challenge was to determine the means to use an overlap-add sinusoidal model effectively for this application. Frame-based models are difficult to apply to signal modification; this is due to the isolated nature of frames and the resulting difficulty in accounting for complex signal dynamics when producing a modified signal.

After studying the modulation effects inherent in overlap-add sinusoidal modeling and quantifying those effects in the frequency domain, it was found that a quasi-harmonic overlap-add model with component frequencies adjusted to preserve intra-frame phase coherence was capable of performing artifact-free modifications, and could do so with minimal computational overhead. Given this refined modifi-

cation model, the fully developed sinusoidal modeling system, dubbed the *Analysis-by-Synthesis/Overlap-Add* (ABS/OLA) system, was then applied to the problems of time-, frequency- and pitch-scale modification of speech. In addition, it was determined that music signals (which are very well modeled using sinusoids) could be analyzed and synthesized as easily as speech signals, with very pleasing musical results.

Having applied the ABS/OLA system to useful problems in digital signal processing, the remaining question was how well it performed relative to the Sine-wave Transform System in similar applications, and whether any advantages were gained by the new approach. This question was answered first by comparing the amount of computation required by both systems, and then by testing the ABS/OLA system against the STS using both objective and formal subjective measures. The results of this testing clearly demonstrated the success of the ABS/OLA system.

Since the research presented in this thesis will be applied primarily to the areas of speech modification and digital music synthesis, it is important to discuss previous work in these application areas, to provide both an historical framework for the present research and meaningful benchmarks for the performance of the developed signal processing algorithms. Therefore, the following section will survey research into the problem of speech modification using digital computers. A brief summary of digital music synthesis algorithms follows, with a more in-depth discussion of this research field deferred until Chapter 6.

1.3 Approaches to Speech Modification

As mentioned before, linear predictive models of speech have been very popular in the area of low bit-rate speech coding, since their parameters correspond to important parameters of the speech production process. From the above discussion of speech modification, however, it is expected that such models would be useful for modifying

speech as well. In linear predictive modeling, speech production is represented by the convolution of a linear time-varying, all-pole *vocal tract filter* with an *excitation signal*. While many variations on this basic model have been used in speech coding [10, 11, 12, 13, 14], the formulation known as *pitch-excited LPC* has been very popular for speech synthesis and modification as well.

In pitch-excited LPC, the excitation signal is modeled either as a pulse train for voiced speech or as white noise for unvoiced speech. This spectrally flat excitation is then shaped by the slowly-varying vocal tract filter, which incorporates characteristics of the glottal source, vocal tract resonance, and radiation effects [15]. The parameters of fundamental frequency and voicing state, and the linear prediction coefficients which determine the vocal tract filter, are determined at fixed intervals in time from measurements of the speech waveform; Figure 1.1 shows a block diagram of the pitch-excited LPC model of speech production.

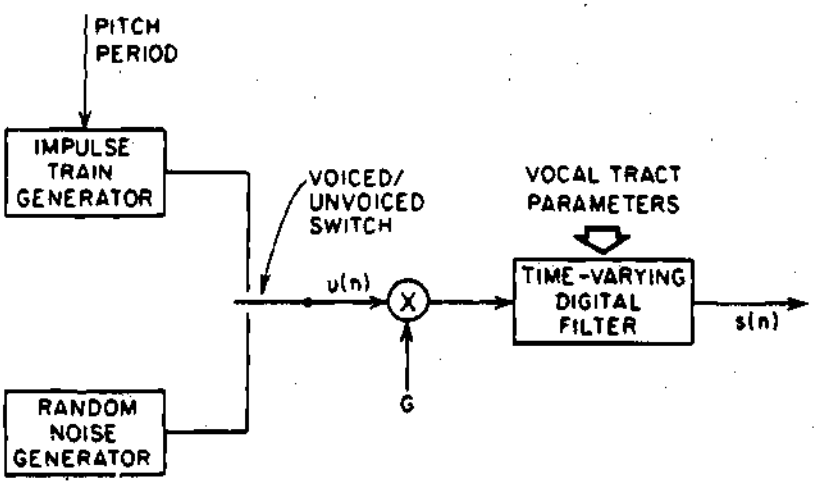

Figure 1.1: Block diagram of pitch-excited LPC model (from [15]).

By effectively separating and parameterizing the voicing state, pitch frequency and articulation rate of speech, pitch-excited LPC can modify analyzed speech in a variety of ways. For instance, the perceived pitch frequency of a speaker may

be modified by changing the measured fundamental frequency to a desired value. Similarly, by changing the rate at which parameters are updated during synthesis, the apparent rate of articulation is altered. In fact, pitch-excited LPC provides enough control over analyzed speech that it is capable of producing artificial speech given linguistic production rules (referred to as *synthesis-by-rule* [16]).

However, pitch-excited LPC is an inherently constrained representation of speech which suffers from well-known distortion characteristics. Linear predictive modeling is based on the assumption that the vocal tract may be modeled using an all-pole filter; depending on how much an actual vocal tract conforms to this ideal assumption, the resulting excitation signal may not possess the purely pulse-like or noisy structure assumed in the excitation model. Pitch-excited LPC therefore produces synthetic speech with a noticeable and objectionable "synthetic" or "buzzy" quality for certain speakers.

Furthermore, linear predictive modeling assumes *a priori* that a given signal is the output of a time-varying filter driven by an easily represented excitation signal, which limits its usefulness to those signals (such as speech) which are reasonably well modeled using this structure. Finally, pitch-excited LPC requires a "voiced/unvoiced" classification and a pitch estimate for voiced speech; unfortunately, the quality of synthetic speech is very sensitive to the accuracy of pitch and voicing state estimation, and serious distortions result from the inevitable errors in both procedures.

The main problems with linear prediction as applied to speech synthesis and modification are related to the difficulty of modeling human speech production. An alternative approach to speech modeling is related to knowledge of the nature of speech signals and of the manner in which human listeners perceive speech. *Time-frequency* representations of speech exploit the observation that speech signals are quasi-periodic, short-time stationary sequences, and attempt to mimic the ear's short-time spectral analysis of audio signals using digital structures [17, 18]. While a variety of time-frequency representations have been formulated, to date the most pop-

ular for the purpose of speech processing has been the *short-time Fourier transform* (STFT) [19]. For a given discrete-time signal $x[n]$, the STFT is defined as

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\omega m}. \tag{1.1}$$

In this formulation, $w[n-m]$ is a real window sequence which slides along the input speech signal, serving to emphasize an interval of $x[m]$ for spectral analysis at time n.

Two interpretations of the STFT are widely used as frameworks in which to define speech modification systems. One interpretation is seen by rewriting Equation 1.1 as

$$X(n,\omega) = e^{-j\omega n} \sum_{m=-\infty}^{\infty} x[n-m]w[m]e^{j\omega m}. \tag{1.2}$$

Examining this expression, for fixed ω the STFT may be viewed as the result of convolving $x[n]$ with a filter whose impulse response is given by $w[n]e^{j\omega n}$, and modulating the result by $e^{-j\omega n}$. If $w[n]$ corresponds to a lowpass filter with a bandwidth of $2\pi/N$ (or an approximate lowpass filter under certain conditions [20]), and the STFT is sampled in the frequency domain at frequencies of

$$\omega_k = \frac{2\pi k}{N},$$

then the resulting *discrete short-time Fourier transform* (DSTFT) [21], $X[n,k]$, can be viewed as the output of a bank of bandpass digital filters, each with an impulse response of

$$h_k[n] = w[n]e^{j\omega_k n};$$

Figure 1.2 illustrates the structure just defined.

This paradigm, referred to as the *filter bank interpretation* of the DSTFT, forms the basis of the *digital phase vocoder* (DPV) [22]. In order to understand the operation of the DPV, consider the case when $x[n]$ corresponds to a sum of complex exponential components with time-varying amplitude and offset phase functions;

$$x[n] = \sum_{i=0}^{N-1} A_i[n]e^{j\{(2\pi/N)in+\Phi_i[n]\}}$$

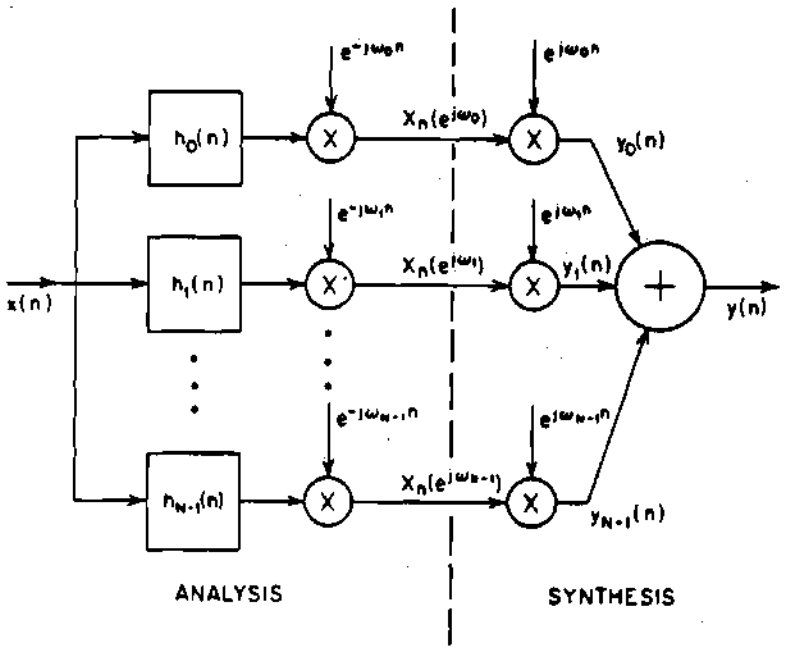

Figure 1.2: Illustration of the filter bank interpretation of the DSTFT (from [19]).

$$= \sum_{i=0}^{N-1} (A_i[n]e^{j\Phi_i[n]})e^{j(2\pi/N)in},$$

where the bandwidth of $A_i[n]e^{j\Phi_i[n]}$ is less than $2\pi/N$ for all i. Referring to Figure 1.2 and assuming an ideal filter bank, the i-th component in this sum is passed unaltered by the i-th filter and rejected by all others. Recalling the derivation of the filter bank interpretation, after modulating the filter outputs by $e^{-j(2\pi/N)kn}$ the resulting DSTFT is

$$X[n,k] = A_k[n]e^{j\Phi_k[n]}, \quad 0 \leq k < N. \tag{1.3}$$

In this example, for a given value of k the DSTFT is seen to correspond to the slowly-varying amplitude and phase offset functions of the k-th component, where

$$\begin{aligned} A_k[n] &= |X[n,k]| \\ \Phi_k[n] &= \angle X[n,k]. \end{aligned} \tag{1.4}$$

It is these amplitude and phase functions which are manipulated in the phase vocoder to modify analyzed speech. In the DPV, phase information embedded in $\Phi_k[n]$ is processed in terms of a *phase derivative* sequence $\Omega_k[n]$, which is often approximated by

$$\Omega_k[n] = \Phi_k[n] - \Phi_k[n-1],$$

where $\Phi_k[n]$ is first "unwrapped" to remove 2π phase ambiguities, thus avoiding discontinuities in $\Omega_k[n]$. If N is sufficiently large that only one harmonic of the speech signal falls in the passband of a given filter at a given point in time, it can be argued that the amplitude functions $\{A_k[n]\}$ correspond to the slowly varying amplitude frequency response of the vocal tract, and that the phase derivative functions $\{\Omega_k[n]\}$ provide information about the time evolution of the excitation [23].

Given this parameterization of speech production in terms of phase vocoder parameters, it is possible to perform a variety of useful speech modifications. For instance, *frequency-scale modification*, or alteration of the frequency dimension of

analyzed speech without corresponding change in the rate of articulation, may be accomplished by scaling the "instantaneous frequency" of each component,

$$\frac{2\pi k}{N} + \Omega_k[n]$$

by a factor β. Conversely, if frequency-scale modified speech is played back at $1/\beta$ speed, then the rate of articulation of analyzed speech is scaled by $1/\beta$ without altering its frequency content; this is known as *time-scale modification.*

Unfortunately, frequency-scale modification of speech using the DPV alters component frequencies without changing component amplitudes, resulting in a compressed or expanded short-time *spectral envelope* for the modified speech. While this may be desirable in applications such as bandwidth compression for the hearing impaired [24], for purposes of altering the pitch of an analyzed speaker the resulting loss of intelligibility and identifiability is very undesirable.

This effect may be counteracted by noting that the frequency response of the vocal tract filter derived from LPC analysis, corresponds to an estimate of the spectral envelope of $X(n,\omega)$ [13]. Given this *spectral envelope estimate*, it is possible to alter component amplitudes in the presence of frequency modification such that the fundamental frequency of analyzed speech is changed while its original formant structure remains intact [25]; this process is referred to as *pitch-scale modification.* Of course, other types of spectral envelope estimates such as those based on smoothing $|X(n,\omega)|$, or homomorphic signal processing [26], may be used in place of the LPC estimate. Henceforth in this work, spectral envelope estimates will be denoted as $H(e^{j\omega})$.

The most significant advantage of using the DPV (or any time-frequency representation, for that matter) to perform speech modifications is that while speech production parameters are accounted for in modeling and modification, the quality of synthetic and modified speech is not contingent on explicit approximation of these parameters, and is thus relatively insensitive to the accuracy or appropriateness of such an approximation. As a result, speech synthesized and modified using the DPV

generally sounds very natural and artifact-free, unlike synthetic speech generated using pitch-excited LPC.

Unfortunately, to generate the amplitude and phase functions used in the phase vocoder it is necessary to implement $N/2$ digital filters[2] at the same rate as the input speech. As a result, the DPV is computationally intensive, limiting its usefulness in many applications. However, certain observations are useful in reducing the computational load associated with the DPV; for instance, noting that $X[n,k]$ is the output of a lowpass digital filter with cutoff frequency π/N, $X[n,k]$ may be downsampled in time by a factor of N without information loss, reducing the computational load required to determine $X[n,k]$.

One approach to further reducing computation in the context of speech modification may be seen by referring to the previous discussion of DPV behavior, and to Equation 1.3. In that discussion, phase vocoder analysis of a slowly-varying quasi-periodic sequence with fundamental frequency $2\pi/N$ was argued to produce amplitude and phase functions which may then be altered to yield desired modifications. Malah [27] has used this observation to define a pitch-adaptive structure for the DPV, where the filter bank frequencies are constrained to be multiples of an estimated fundamental frequency.

While such a structure would be unwieldy if implemented directly, Malah has shown that this pitch-adaptive phase vocoder may be approximated in the time domain by manipulating windowed segments of $x[n]$ in a pitch-synchronous fashion. This approach, known as *Time-Domain Harmonic Scaling* (TDHS), produces very high quality time- and frequency-scale modified speech given accurate pitch estimates, and does so at a computational load of one multiplication and two additions per sample. The main problem with TDHS is that, like pitch-excited LPC, modified speech quality is very sensitive to pitch estimation errors; nevertheless, TDHS remains a popular algorithm for real-time speech modification due to its simplicity.

[2]Instead of N filters, due to symmetry properties when $x[n]$ is real.

Another approach to reducing the computational load of the DPV is derived by making use of the *Fourier transform interpretation* of the downsampled DSTFT, given from Equation 1.1 as

$$X[nN,k] = \sum_{m=-\infty}^{\infty} x[m]w[nN-m]e^{-j(2\pi/N)km}. \tag{1.5}$$

It can be shown that by time-aliasing the sequence $x[nN-m]w[m]$, $X[nN,k]$ may be viewed as a sequence of *discrete Fourier transforms* (DFT's) calculated at intervals of N samples in time using the FFT algorithm, at a significant computational savings [28]. Furthermore, it can also be shown that $x[n]$ is recoverable from $X[nN,k]$ by overlapping and adding sequences derived from the downsampled DSTFT of Equation 1.5 (a process referred to as *overlap-add synthesis* [29, 30]), and that these sequences may be computed using the FFT algorithm [31].

Using the DSTFT formulation just described, Portnoff has proposed a system for performing time- and frequency-scale modification of speech [32]. In this system, the phase of synthetic speech is derived by approximating the phase behavior of the DPV, using stationarity arguments and inter-frame phase continuity constraints. While this approach takes advantage of the computational efficiency gained by using the FFT for implementation and provides much of the functionality of the DPV, the approximate nature of synthetic phase causes problems in speech modification. Specifically, approximation of the phase behavior of the DPV in the presence of modifications results in the propagation of phase offsets in components used for synthesis; these phase offsets cause the response of the overall analysis/synthesis system to deviate from the desired "flat" response with linear phase, resulting in modified speech with a distinct reverberant quality [19].

In an attempt to deal with phase problems encountered in Portnoff's approach, Griffin and Lim [33] have proposed an algorithm designed to estimate optimal phase for a time-scale modified sequence $\hat{x}[n]$ such that the STFT magnitude of $\hat{x}[n]$ is as close as possible to the time-scaled STFT magnitude of $x[n]$. This algorithm is based on the calculation of a sequence $\hat{x}[n]$ whose STFT is as close as possible to a desired

STFT, $Y(nN,\omega)$, using the error norm

$$\sum_{r=-\infty}^{\infty} \frac{1}{2\pi} \int_{-\pi}^{\pi} [\hat{X}(rN,\omega) - Y(rN,\omega)]^2; \tag{1.6}$$

The optimal sequence $\hat{x}[n]$ may be calculated as

$$\hat{x}[n] = \frac{\sum_{r=-\infty}^{\infty} w[rN-n]y[rN,n]}{\sum_{r=-\infty}^{\infty} w^2[rN-n]}, \tag{1.7}$$

where $y[rN,n]$ is the inverse transform of $Y(rN,\omega)$ for a given value of r. This approximation is necessary because $Y(nN,\omega)$ is not, in general, a valid STFT. In other words, given an arbitrary $Y(nN,\omega)$, there is no guarantee that a sequence exists whose STFT is $Y(nN,\omega)$. Equation 1.7, then, calculates the sequence whose valid STFT is closest to $Y(nN,\omega)$ in the sense of Equation 1.6.

In the context of time-scale modification, the modified STFT magnitude $|Y(nN,\omega)|$ is derived from $X(nN,\omega)$ by interpolating $|X(nN,\omega)|$ to have the desired time scale. Since no phase information is now available in the modified STFT, it is necessary to generate this information iteratively. This is accomplished as follows: Given an estimated signal at the i-th iteration, $\hat{x}^i[n]$, the STFT of this sequence, $\hat{X}^i(nN,\omega)$ is calculated. An i-th modified STFT is constructed which has the desired magnitude but the phase of $\hat{X}^i(nN,\omega)$, *i.e.*,

$$Y^i(nN,\omega) = |Y(nN,\omega)|e^{j\angle \hat{X}^i(nN,\omega)}.$$

The inverse transform of $Y^i(rN,\omega)$, $y^i[rN,n]$, is then substituted into Equation 1.7 to generate $\hat{x}^{i+1}[n]$, and the process is repeated for the next iteration. This algorithm, known as the *Least-Squares Error Estimation from the Modified Short-Time Fourier Transform Magnitude* (LSEE-MSTFTM) algorithm, was reported to generate high-quality time-scale modified speech after approximately 25 iterations, and largely eliminates the reverberant artifact associated with Portnoff's method. In addition, other modifications such as pitch-scale modification and formant shifting are possible by specifying other desired STFT magnitudes [34].

Unfortunately, the LSEE-MSTFTM algorithm requires nearly as much computation to implement as the DPV, due to the large number of iterations required to converge on a reasonable solution. The source of these computational requirements can be traced in large part to the initial estimate of $\hat{x}[n]$ used to initialize the algorithm. In Griffin and Lim's work, $\hat{x}^0[n]$ is initially a Gaussian random sequence. Random initialization, while more general in form, ignores much of the information embedded in speech signals which may be used to form a better initial estimate.

Roucos and Wilgus have formulated an algorithm which attempts to find a more meaningful initial estimate for the LSEE-MSTFTM algorithm [35]; in their approach, the formula of Equation 1.7 is used to determine $\hat{x}^0[n]$, with the sequences $y[rN, n]$ substituted with windowed segments of the original speech signal which are shifted to maximize cross-correlation. The *Synchronized Overlap-Add* (SOLA) algorithm significantly reduces the number of iterations required to generate high-quality synthetic speech; in fact, reasonable quality time-scale modification with this technique is possible without using the LSEE-MSTFTM algorithm at all, resulting in a modification system with computation comparable to TDHS, but without the attendant need for, or sensitivity to, pitch estimation.

1.3.1 Sinusoidal Modeling

As discussed in the previous section, the advantage of speech models based on short-time Fourier analysis over linear predictive models is the robustness gained in speech synthesis and modification by using a less constrained, more general representation of speech production parameters. However, the discussion above also makes it clear that, in the case of the phase vocoder with a fixed number of frequencies, the relation of speech production parameters to model parameters is indirect and at times tenuous. Furthermore, referring to the discussion of Portnoff's method, limited time and frequency resolution in the parameterization of speech using computationally tractable formulations of the DSTFT can cause significant problems in the presence

of modifications.

An approach to further generalizing time-frequency parameterizations of speech was suggested by Hedelin [36], who used a pitch-independent sinusoidal representation of speech excitation in the baseband for the purposes of medium bit-rate speech coding. His work was based on the idea that speech signals may be represented directly using amplitude- and frequency-modulated sinusoids which reflect the pitch structure of voiced speech and the formant structure and random character of unvoiced speech. Similar ideas were developed by Almeida and Silva for mid-rate coding using harmonic models [37].

The notion of representing speech using sinusoidal signals was developed in a more general framework by McAulay and Quatieri [4], whose work introduced a variety of techniques for dealing with the estimation and modeling problems encountered in sinusoidal modeling, resulting in the aforementioned Sine-wave Transform System. For instance, given a sequence composed of a sum of sinusoids as

$$s[n] = \sum_l A_l \cos(\omega_l n + \theta_l), \tag{1.8}$$

it can be shown that the component frequencies of $s[n]$ are approximately given by the location of magnitude peaks of the spectrum of $w_a[n]s[n]$, where $w_a[n]$ is a symmetric, tapered *analysis window* such as a Hamming window whose spectrum approximates a frequency-domain impulse [38], and that reasonable estimates of the amplitude and phase parameters of each component are derived from complex spectral values at the corresponding frequencies.

On this basis, McAulay and Quatieri developed an analysis algorithm using an alternate formulation of the STFT[3] given by

$$\overline{X}(n,\omega) = \sum_{m=-N_a}^{N_a} w_a[m]x[m+n]e^{-j\omega m} = e^{j\omega n}X(n,\omega). \tag{1.9}$$

Assuming that (1) the analysis window $w_a[m]$ is short enough that the windowed portion of $x[m+n]$ can be assumed to be short-time stationary and (2) $w_a[m]$ is long

[3]Assuming a symmetric, finite-length analysis window.

enough to resolve the component frequencies of the speech segment, the sinusoidal model parameters which approximately represent $s[n]$ over time may be calculated every N_s samples in time from the alternate DSTFT

$$\overline{X}[kN_s, i] \triangleq \overline{X}(kN_s, (2\pi/I)i)$$

by locating magnitude peaks of $|\overline{X}[kN_s, i]|$ to determine frequency estimates $\{\hat{\omega}_l^k\}$ and sampling $\overline{X}[kN_s, i]$ at peak locations to determine amplitudes $\{\hat{A}_l^k\}$ and phases $\{\hat{\theta}_l^k\}$.

The DFT length I is typically much larger than N_a to ensure accurate component frequency estimation, and $\overline{X}[kN_s, i]$ is calculated using the FFT algorithm by zero-padding $w_a[m]x[m + kN_s]$ to length I. To provide reasonably good frequency resolution for a wide range of pitch frequencies, N_a is adapted to the average pitch frequency of a given speaker to provide an analysis window length of 2.5 pitch periods [4]. The spectrum of a typical voiced speech segment, and the amplitude and frequency estimates derived from its magnitude peaks, is shown in Figure 1.3.

Given sinusoidal model parameters determined from this *peak-picking* algorithm, a straightforward method for synthesizing speech is simply to overlap and add window-weighted signals generated using the analyzed parameters, according to the relation

$$\tilde{s}[n] = \sum_{k=-\infty}^{\infty} w_s[n - kN_s]\tilde{s}^k[n - kN_s], \tag{1.10}$$

where

$$\tilde{s}^k[n] = \sum_l \hat{A}_l^k \cos(\hat{\omega}_l^k n + \hat{\theta}_l^k), \tag{1.11}$$

using a suitable complementary window function (such as a triangular window or Hanning window) for $w_s[n]$. While this strategy succeeds for synthesis frame lengths on the order of 10 msec, synthetic speech quality seriously degrades for longer frame lengths due to violation of the stationarity assumption. In order to use longer frame lengths (and hence less computation), and to provide a functional framework for speech synthesis and modification, McAulay and Quatieri instead formulated a syn-

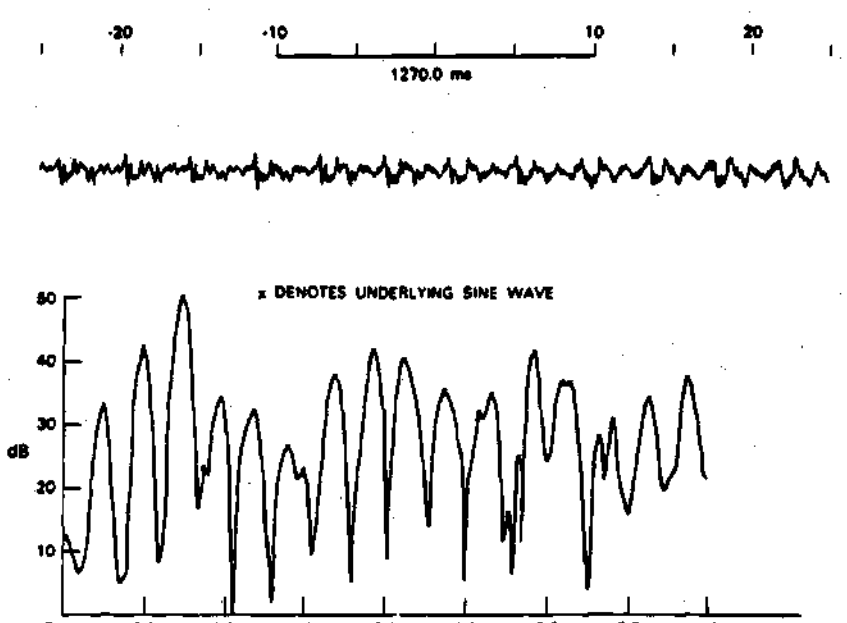

Figure 1.3: Example of the peak-picking analysis algorithm of McAulay and Quatieri used for sinusoidal modeling (from [4]).

thesis model for speech based on parameter interpolation over time using piecewise continuous polynomial functions.

To perform parameter interpolation, it is necessary to match sets of parameters from one frame to the next; furthermore, since the pitch and spectral content of speech changes over time, it is necessary to deal with changing numbers of components in the proposed synthesis model. Both of these goals are achieved in the Sine-wave Transform System by using a simple "nearest-neighbor" frequency-matching algorithm. In this algorithm, frequencies determined from peak picking at frames k and $(k+1)$ are compared; given a frequency ω_l^k from frame k, the closest unmatched frequency ω_m^{k+1} in frame $k+1$ is determined. If ω_m^{k+1} is within a "matching interval" Δ of ω_l^k, and ω_m^{k+1} is not closer to another unmatched frequency in frame k, then the two frequencies are matched. This process is repeated until no valid pairings are available.

The nearest neighbor algorithm produces related frequencies over a series of frames which form a "frequency track." Due to the nature of the algorithm, some frequencies in frame k or $k+1$ may not be matched to frequencies in adjacent frames, corresponding to the "death" and "birth," respectively, of frequency tracks; this behavior is shown in Figure 1.4(a). Given matched parameter sets from one frame to the next, amplitude functions are generated by linear interpolation according to

$$\hat{A}[n] = \hat{A}^k + \frac{(\hat{A}^{k+1} - \hat{A}^k)}{N_s} n, \quad 0 \leq n < N_s; \tag{1.12}$$

in order to avoid discontinuous parameter tracks, unmatched components are interpolated to zero amplitude across the synthesis frame, shown by dotted lines in Figure 1.4(a). Phase functions are determined by a piecewise cubic interpolator of the form

$$\hat{\theta}[n] = a + bn + cn^2 + dn^3 \tag{1.13}$$

designed to match boundary phase and frequency constraints. An additional complication in phase interpolation is the need to "unwrap" phase as in the phase vocoder; this is accomplished in the Sine-wave Transform System by calculating a multiple of

2π which, when added to $\hat{\theta}_l^{k+1}$, results in an interpolated phase function with minimal curvature. This unwrapping process is illustrated in Figure 1.4(b). Given interpolated amplitude and phase parameters, synthetic speech is given by

$$\tilde{s}[n] = \sum_l \hat{A}_l[n] \cos \hat{\theta}_l[n]. \tag{1.14}$$

Since synthesis in the STS is based on parameter tracks possessing a functional form, time- and frequency-scale modification can be performed in a manner similar to the DPV [5]. Specifically, the amplitude and phase functions given above may be associated with underlying continuous functions $\hat{A}(t)$ and $\hat{\theta}(t)$ simply by replacing the discrete variable n with t. Thus it is possible to alter the rate of synthetic speech by an arbitrary factor $1/\rho$ without changing pitch by time-scaling $\hat{A}(t)$ and $\hat{\omega}(t)$, the phase derivative of $\hat{\theta}(t)$, according to the relations

$$\begin{aligned} \hat{A}'(t') &= \hat{A}(t'/\rho) \\ \hat{\omega}'(t') &= \hat{\omega}(t'/\rho), \end{aligned}$$

where t' denotes the "warped" time variable. Modified synthetic phase is produced from the time-scaled frequency track by symbolic integration, yielding

$$\hat{\theta}'(t') = a' + bt' + \frac{c}{\rho}(t')^2 + \frac{d}{\rho^2}(t')^3,$$

where the offset phase a' is adjusted to maintain phase continuity at frame boundaries. These modified functions are then sampled at integer values of t' and used in Equation 1.14 to generate modified synthetic speech. A similar process may be used to modify the frequency scale of synthetic speech, simply by scaling $\hat{\omega}(t)$ by a factor β.

Unfortunately, since phase parameters in the Sine-wave Transform System are calculated at intervals of N_s samples, this simple approach to modification results in phase error propagation similar to that found in Portnoff's approach, resulting in reverberant modified speech. However, noting that this effect can be viewed as a breakdown in phase coherence due to independent parameter modification for each component, and making use of the source/filter model for speech, McAulay and Quatieri

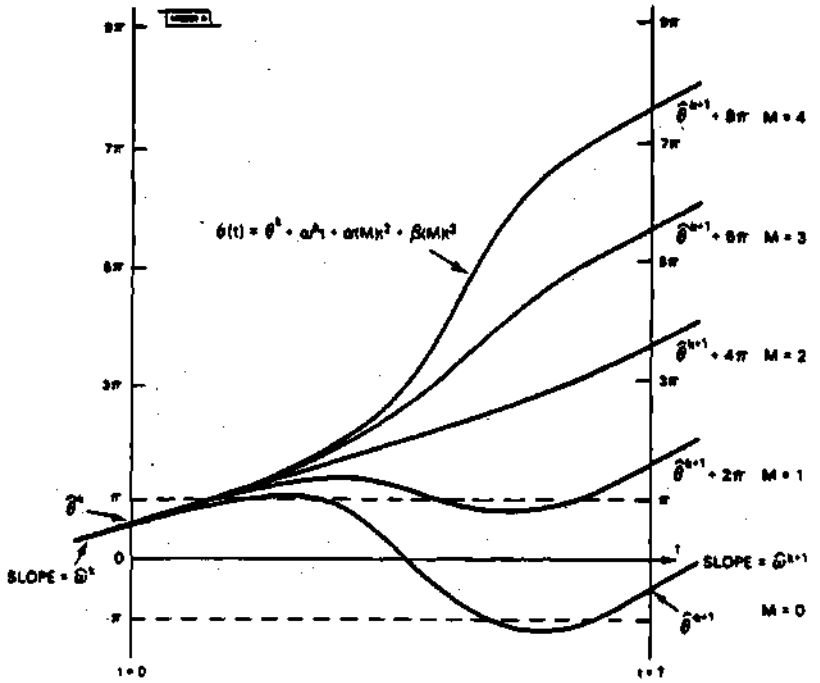

Figure 1.4: Demonstration of phase interpolation process in the Sine-wave Transform System (from [4]).

developed a strategy for speech modification using a *pitch onset time excitation model* based on sinusoidal model parameters [39].

In this model, given a spectral envelope estimate $H^k(e^{j\omega})$ of the speech signal at $n = kN_s$, the underlying excitation sequence in the k-th synthesis frame may be expressed as (ignoring frame notation),

$$\begin{aligned} \tilde{e}[n] &= \sum_l \hat{a}_l \cos(\hat{\omega}_l n + \hat{\phi}_l), \\ &= \sum_l \hat{a}_l \cos(\hat{\omega}_l [n - t_o] + \hat{\psi}_l(t_o)) \end{aligned} \tag{1.15}$$

where

$$\begin{aligned} \hat{a}_l &= \hat{A}_l / |H(e^{j\hat{\omega}_l})| \\ \hat{\phi}_l &= \hat{\theta}_l - \angle H(e^{j\hat{\omega}_l}) \\ \hat{\psi}_l(t_o) &= \hat{\phi}_l + \hat{\omega}_l t_o. \end{aligned} \tag{1.16}$$

According to the source/filter model for voiced speech, $\tilde{e}^k[n]$ should approximately correspond to a sequence of impulses separated by the pitch period of speech in frame k. Equation 1.15 parameterizes this behavior in terms of the "pitch onset time" t_o at which a pulse occurs relative to $n = kN_s$. Under ideal conditions, at $n = t_o$ the components of $\tilde{e}^k[n]$ will add coherently, implying that residual phase parameters $\{\hat{\psi}_l(t_o)\}$ will all equal zero or π. Thus, t_o can be estimated by calculating the likelihood function

$$\ell(t_o) = \sum_l \hat{a}_l^2 \cos \hat{\psi}_l(t_o) = \sum_l \hat{a}_l^2 \cos(\hat{\phi}_l + \hat{\omega}_l t_o) \tag{1.17}$$

for candidate values of t_o, choosing the value corresponding to the absolute maximum of $\ell(t_o)$ [39].

The result of pitch onset time estimation is knowledge of the location of pitch pulses of unmodified speech. For the purpose of speech modification, this information is combined with fundamental frequency values from frame to frame and exploited in the Sine-wave Transform System to maintain the relationship of pitch pulse locations in the presence of speech modification, a technique referred to as *shape invariant*

modification [40]. To understand this process in the context of time-scale modification, consider the case of a segment of voiced speech covering several synthesis frames, shown in Figure 1.5; the pitch onset time of unmodified speech relative to frame boundary T is shown in the upper plot. The pitch pulse at this location is located approximately a pitch period from an adjacent pulse located in the previous frame.

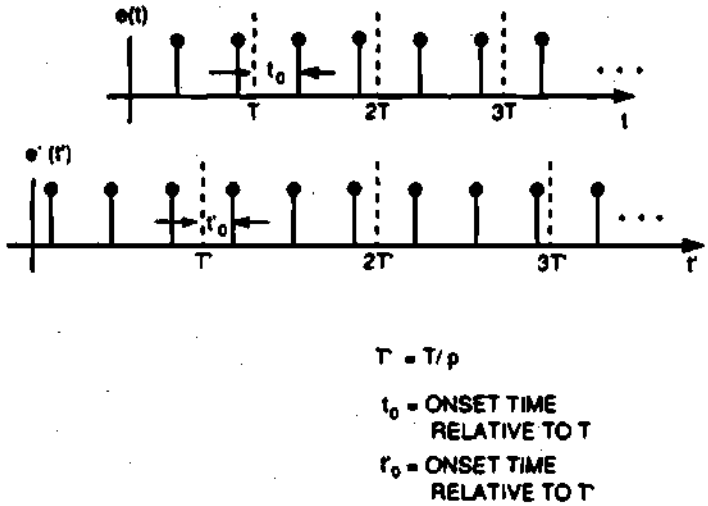

Figure 1.5: Illustration of shape-invariant time-scale modification algorithm (from [40]).

In the presence of time-scale modification, pitch pulse separations are unchanged, but synthesis frame lengths are altered by a factor of ρ. Therefore, it is necessary to introduce a time shift to the excitation sequence in each modified frame to ensure that the distance between pulses across modified frame boundaries (such as T' in the lower plot) are the same as across unmodified frame boundaries. As seen in the lower plot, this results in a modified pitch onset time t'_o for the frame of interest. Since time shifts introduced to underlying excitation sequences $\tilde{e}^k[n]$ correspond to similar time shifts of $\tilde{s}^k[n]$, time-scale modification may be implemented in the STS by introducing

a shift of $\delta^k = (t'_o - t_o)$ to $\tilde{s}^k[n]$, yielding modified synthetic phases of

$$\hat{\theta}'_l = \hat{\theta}_l + \hat{\omega}_l \delta^k.$$

These modified phase parameters are then separated in time by ρN_s samples and used in conjunction with unaltered amplitude parameters $\{\hat{A}_l\}$ and frequencies $\{\hat{\omega}_l\}$ to form interpolated amplitude and phase functions. The modified functions are then used in Equation 1.14 to generate modified speech. As in the previous discussion of modification using the STS, the same strategy may be used for frequency-scale modification by scaling pitch pulse spacing by $1/\beta$ and frequency parameters by β.

Shape-invariant modification, as the name implies, is capable of producing a modified speech signal whose waveform maintains the structure of the original speech, and largely eliminates phase distortion associated with other time-frequency approaches to speech modification. The reason for this level of performance is actually quite simple: Since a global time shift is applied to $\tilde{s}^k[n]$ in each synthesis frame to account for temporal phase continuity, the original phase relationships between components of $\hat{s}^k[n]$ are preserved reasonably well regardless of the transformation, provided the time shift is not excessive. It is also worth noting that while pitch information is used in shape-invariant modification to determine modified phase parameters, the resulting modification system is not pitch-driven since the frequency parameters used in synthesis are not required to be harmonic.

As in the case of modification using the STFT, it is possible in the Sine-wave Transform System to use a time-varying spectral envelope estimate of the speech signal to perform pitch-scale modification. Generally speaking, this is accomplished by generating parameters for the speech excitation using $H^k(e^{j\omega})$ and the formula of Equation 1.15, performing frequency-scale modification of the resulting excitation signal, then generating modified amplitude and phase parameters for the synthesis model by reversing the operations of Equation 1.16. However, a serious problem with this strategy is that most popular methods for spectral envelope estimation assume, for the sake of simplicity, that the vocal tract is a *minimum-phase* system [41]; this

assumption typically results in an excitation sequence with considerably dispersed pitch pulses. In the presence of frequency-scale modification, this dispersion is increased or decreased depending on β, resulting in inconsistent quality in pitch-scale modification.

To deal with this problem, McAulay and Quatieri have formulated a method for generating a "mixed-phase" spectral envelope estimate within the context of the pitch onset time excitation model, based on $H(e^{j\hat{\omega}_l})$ and analyzed sinusoidal model parameters [42]. Referring to Equation 1.16, $\hat{\theta}_l$ may be expressed as

$$\begin{aligned}\hat{\theta}_l &= \hat{\phi}_l + \angle H(e^{j\hat{\omega}_l}) \\ &= -\hat{\omega}_l t_o + (\angle H(e^{j\hat{\omega}_l}) + \hat{\psi}_l(t_o)),\end{aligned}$$

so that

$$\hat{\theta}_l - (\angle H(e^{j\hat{\omega}_l}) + \hat{\psi}_l(t_o)) = -\hat{\omega}_l t_o.$$

Using this result, the mixed-phase spectral estimate $H_{\mathrm{mp}}(e^{j\omega})$ may be defined at component frequencies $\{\hat{\omega}_l\}$ by

$$H_{\mathrm{mp}}(e^{j\hat{\omega}_l}) \triangleq |H(e^{j\hat{\omega}_l})| e^{j(\angle H(e^{j\hat{\omega}_l}) + \hat{\psi}_l(t_o))}, \tag{1.18}$$

and at arbitrary frequencies by complex interpolation in frequency of these samples. By substituting $H_{\mathrm{mp}}(e^{j\hat{\omega}_l}))$ into Equation 1.16, the resulting excitation sequence has the form

$$\tilde{e}_{\mathrm{mp}}[n] = \sum_l \hat{a}_l \cos(\hat{\omega}_l[n - t_o]); \tag{1.19}$$

This mixed-phase excitation sequence possesses no pitch pulse dispersion due to phase, thus when used in the pitch-scale modification strategy described above in conjunction with $H_{\mathrm{mp}}(e^{j\hat{\omega}_l})$, the subjective quality of resulting pitch-modified speech is considerably improved.

While the Sine-wave Transform System as described so far succeeds in producing high-quality synthetic speech and performing robust speech modification which eliminates the distortions associated with other approaches to speech modification, it

still has a serious weak spot. Referring to Equation 1.14, the STS synthesis model is seen to be of the same form as that used in the phase vocoder; as a result, this model must be implemented by generating each component sinusoid over time using digital oscillator structures or stored tables of sinusoidal values. The computational load of such an implementation is formidable, and seriously limits the ability to use the STS in a real-time environment.

As mentioned previously, it is possible to implement synthesis with a sinusoidal model using an overlap-add structure for synthesis frame lengths on the order of 10 msec or less; furthermore, in this approach each $\tilde{s}^k[n]$ may be generated using the IFFT algorithm, which dramatically reduces computational requirements. Since shape-invariant speech modification does not require interpolated amplitude and phase parameters *per se*, it is possible to implement the Sine-wave Transform System using overlap-add synthesis given a sufficiently high frame rate. However, to insure a frame length of 10 msec in the context of time-scale modification by a factor of two, for instance, requires analysis at 5 msec intervals, considerably increasing the computation required for analysis.

To deal with this paradox, McAulay and Quatieri have devised a strategy for analyzing speech every N_s samples while performing overlap-add synthesis every $N_s/2$ samples [43]. This strategy is based on estimating sinusoidal model parameters at the middle of a synthesis frame by averaging matched parameter sets in adjacent frames, yielding amplitudes $\{\overline{A}_l\}$, frequencies $\{\overline{\omega}_l\}$ and phases $\{\overline{\theta}_l\}$ which produce a "midframe" synthetic sequence $\overline{s}^k[n]$. Synthetic speech is then produced in a synthesis frame using overlap-add synthesis of the three sequences $\tilde{s}^k[n]$, $\overline{s}^k[n]$ and $\tilde{s}^{k+1}[n]$ with a synthesis window half as long as would otherwise be required. Figure 1.6 illustrates the concept of half-rate synthesis in the STS.

The Sine-wave Transform System represents a high-quality alternative to linear predictive modeling and speech modification using the STFT, and offers advantages over these approaches for synthesis and modification problems. As with the STFT,

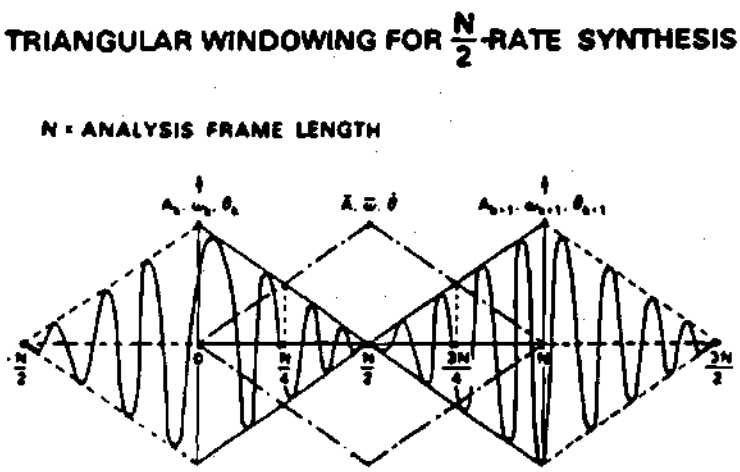

Figure 1.6: Illustration of half-rate overlap-add synthesis in the Sine-wave Transform System (from [43]).

sinusoidal modeling operates without an "all-pole" constraint, which results in natural sounding synthetic and modified speech. Also, sinusoidal modeling does not require the restrictive "source/filter" structure of linear predictive modeling; sinusoidal models are thus capable of representing signals from a variety of sources, including speech from multiple speakers, music signals, speech in musical backgrounds, as well as marine biological and certain biomedical signals. The Sine-wave Transform System parameterizes speech production in a more direct fashion than the STFT, thus providing greater access to and control over speech production parameters than the STFT; the most significant result of this level of control is the ability to deal with phase coherence issues in modification, thus largely eliminating the phase distortion associated with speech modification using the STFT.

While the Sine-wave Transform System represents a significant step forward in the area of speech analysis/synthesis and speech modification, it does have some drawbacks. As discussed previously, peak-picking analysis assumes the optimality of peak parameters, based on stationarity arguments; for non-stationary speech events such as plosives, this assumption tends to result in undesirable smoothing effects. Furthermore, spectral interference effects due to windowing reduce the accuracy of parameter estimates, particularly in the mid- to high-frequency ranges; the result is often a subtle but persistent "tonal" quality to synthetic and modified speech.

As mentioned before, the interpolated parameter synthesis model of the STS departs from the theoretical basis of analysis, significantly so for non-stationary speech events: this further exacerbates the smoothing effect for plosives. While shape-invariant modification is largely successful in avoiding phase problems associated with other time-frequency approaches to modification, it possesses some sensitivity to the time-shifts introduced to preserve phase coherence, depending on the harmonicity or stationarity of analyzed speech; this sensitivity can manifest itself as a reververant quality when slowing fast speakers.

The "mixed-phase" pitch modification algorithm produces high-quality pitch

modified speech, but fails to deal with the effect of bandwidth compression when lowering pitch, resulting in a "muffled" speech quality. Furthermore, the issue of *noise migration* still exists in this approach, *i.e.* noise effects which are not perceptually important in unmodified speech may be amplified when shifted in frequency to a formant region. Finally, while the half-rate overlap-add synthesis algorithm succeeds in reducing computational requirements, it makes no attempt to account for the time-frequency behavior of speech in its implementation, which places limits on the amount of modification possible using this approach. McAulay and Quatieri have reported a loss in reproduction accuracy due to parameter estimation in half-rate synthesis, but interestingly have noted a slight improvement in subjective quality as a result [43].

1.4 Approaches to Music Synthesis

Many techniques for the digital generation of musical sounds have been studied, and many are used in commercially available music synthesizers. In all of these techniques a basic tradeoff is encountered; namely, the conflict between accuracy and generality[4] on the one hand and computational efficiency on the other. Some techniques, such as *frequency modulation* (FM) synthesis [44], are computationally efficient and can produce a wide variety of musically interesting sounds, but lack the ability to accurately model the sounds of existing musical instruments.

On the other hand, *sinusoidal additive synthesis* implemented using the DPV is capable of analyzing the sound of a given instrument, synthesizing a perfect replica and performing a wide variety of modifications. However, as previously mentioned, the amount of computation needed to calculate time-varying sinusoidal components prohibits real-time synthesis using relatively inexpensive hardware [45]. As in the case of time-frequency speech modeling, the computational problems of additive synthesis of musical tones may be addressed by formulating the DPV in terms of the DSTFT

[4]Defined as the ability to model a wide variety of sounds

and implementing this formulation using the FFT algorithm. Unfortunately, this strategy produces the same type of distortion when applied to musical tone synthesis as to speech synthesis.

1.5 Thesis Outline

This thesis presents a new, highly structured analysis/synthesis system for audio signals based on the combination of an overlap-add sinusoidal model formulation and an analysis-by-synthesis procedure which determines appropriate model parameters. The research presented contributes to the area of parametric signal modeling in general and to sinusoidal modeling in particular by investigating the effect of combining overlap-add sinusoidal modeling with analysis-by-synthesis, and by assessing the impact of this analysis/synthesis system for speech and music signal processing applications. In particular, the following topics will be addressed:

1. Introduction of a sinusoidal analysis/synthesis system based on analysis-by-synthesis, and detailed discussion of issues related to its operation.

2. Development of a generalized overlap-add sinusoidal model which allows for distortion-free modification of speech and music signals.

3. Application of the above system to the problems of speech analysis/synthesis, speech modification and music synthesis.

4. Development of techniques to reduce the computational load involved in analysis and synthesis procedures.

5. Testing and evaluation of the developed system relative to similar results generated using the Sine-wave Transform System.

The next chapter presents a mathematical treatment of an iterative technique for approximating vectors in finite-dimensional vector spaces, referred to as *iterative*

vector approximation. Chapter 3 describes the application of iterative vector approximation in an analysis-by-synthesis procedure to the problem of determining the parameters of an overlap-add sinusoidal model formulation. Chapter 4 derives the refined overlap-add model formulation required for the purposes of modification, and describes the application of this new model to the problems of time-, frequency- and pitch-scale modification. Chapter 5 derives several relationships between equations in the analysis and synthesis algorithms and the *discrete Fourier transform* (DFT), which can be used to significantly reduce the required computational load. Chapter 6 details the use of the developed system in the analysis, synthesis and modification of musical tones. Chapter 7 compares the performance of the analysis/synthesis system presented with that of the Sine-wave Transform System of McAulay and Quatieri, using both objective and subjective measures. Finally, Chapter 8 concludes the thesis by reviewing the research results presented, interpreting the results to determine their significance, and discussin possible future directions for research on this topic.

CHAPTER 2

Iterative Vector Approximation

Mathematical representations of discrete-time signals are often derived as the solutions to finite-dimensional vector space problems; LPC filter parameters, for instance, are determined by minimizing a mean-square prediction error norm in terms of the prediction filter coefficients. Similarly, the discrete Fourier transform may be viewed as a simple change of basis from Cartesian coordinates to a basis composed of orthogonal complex exponentials. Such *signal space* formulations are desirable since they can lead to tractable, closed-form, often highly efficient solutions in terms of sets of linear equations, and are easily analyzed in terms of the large body of results from vector space theory and linear algebra.

There are, however, many interesting signal representations whose vector space formulations lead to intractable problems. One example is the *multiple-pulse excited LPC* (MPLPC) model introduced by Atal and Remde for low bit-rate speech coding [9]. The MPLPC (or "multipulse") model may be viewed as a weighted sum of vectors which approximate a given "signal" vector, and can thus be formulated in terms of a least-squares approximation problem to determine optimal weighting coefficients. Unfortunately, the vector set used in the multipulse model typically has too many members to be linearly independent. Therefore, an unambiguous least-squares approximation cannot be determined using standard analytical approaches.

In order to provide a tractable solution to the MPLPC problem, Atal and Remde employed an analysis-by-synthesis procedure which incorporates an iterative, search-based approach to vector approximation. This approach, which has since been applied

successfully to other linear prediction-based vocoders [11, 12], also forms the basis of the analysis procedure presented in this thesis. This chapter thus introduces *iterative vector approximation* as an approach to solving a non-unique vector approximation problem in a real, finite-dimensional vector space, details the concepts and formulas associated with the technique, and analyzes its properties in terms of vector space theory.

2.1 General Concepts and Formulas

Suppose we are given the p-dimensional Cartesian vector space $\boldsymbol{R}^p$ of real ordered "p-tuples," where for a vector $\mathbf{x} \in \boldsymbol{R}^p$,

$$\mathbf{x} = (x_1, x_2, \ldots, x_p)^T. \tag{2.1}$$

This vector space is an *inner product space*, where the inner product $\langle \mathbf{x}, \mathbf{y} \rangle$ between vectors $\mathbf{x}$ and $\mathbf{y}$ is defined as

$$\langle \mathbf{x}, \mathbf{y} \rangle \triangleq \sum_{i=1}^{p} x_i y_i. \tag{2.2}$$

and where the *Euclidean norm* of $\mathbf{x}$ is given by

$$\|\mathbf{x}\| \triangleq \langle \mathbf{x}, \mathbf{x} \rangle^{1/2} = \left(\sum_{i=1}^{p} x_i^2 \right)^{1/2}. \tag{2.3}$$

Now consider the following problem: Given a vector $\mathbf{x} \in \boldsymbol{R}^p$, an approximation to $\mathbf{x}$ of the form

$$\tilde{\mathbf{x}} = \sum_{j=1}^{J} \hat{\mathbf{x}}_j \tag{2.4}$$

is to be constructed such that $\tilde{\mathbf{x}}$ is "close" in some sense to $\mathbf{x}$. The *components* of the approximation are given by

$$\hat{\mathbf{x}}_j = \sum_{k=1}^{K} a_k^j \mathbf{v}_k^j. \tag{2.5}$$

The vectors $\mathbf{v}_k^j$ are drawn from K *vector ensembles*, each a set of I vectors indexed from 1 to I. Thus, the approximation vector $\tilde{\mathbf{x}}$ is composed of vectors $\{\hat{\mathbf{x}}_1, \ldots, \hat{\mathbf{x}}_J\}$,

and each of these component vectors is itself a linear combination of the K vectors $\{\mathbf{v}_1^{i_j}, \ldots, \mathbf{v}_K^{i_j}\}$ drawn from the vector ensembles at a fixed index value i_j. It is worth noting that the sequence of indices $i_1, \ldots, i_j$ is not required to possess any structure: however, if unique values of each index are desired, J should clearly be less than or equal to I.

The most common approach to determining appropriate values for the parameters of $\tilde{\mathbf{x}}$ is to minimize the squared error norm

$$E = \|\mathbf{e}\|^2 = \|\mathbf{x} - \tilde{\mathbf{x}}\|^2 \tag{2.6}$$

in terms of these parameters. Unfortunately, two problems arise with this approach. First, if $J < I$, then the set of *ensemble indices* $\{i_j\}$ must be determined as well as the set of *weighting coefficients* $\{a_k^j\}$; this is a very difficult task to accomplish formally. Second, given a fixed set of ensemble indices, when $JK > p$ the *replacement lemma* of linear algebra states that the approximating vectors must form a linearly dependent set: as mentioned previously, in this case the error norm of Equation 2.6 cannot be uniquely minimized in terms of $\{a_k^j\}$ using analytical techniques.

In either of these circumstances, straightforward approaches to error minimization fail to provide a satisfactory solution to the problem. However, given that $K \leq p$ it is possible to solve for the components of $\tilde{\mathbf{x}}$ in an iterative fashion. Iterative vector approximation is formally described as follows: Suppose that the parameters of $\ell - 1$ components have been determined previously, yielding a *sequential approximation vector*,

$$\tilde{\mathbf{x}}_{\ell-1} = \sum_{j=1}^{\ell-1} \hat{\mathbf{x}}_j, \tag{2.7}$$

and a *sequential error vector*

$$\mathbf{e}_{\ell-1} = \mathbf{x} - \tilde{\mathbf{x}}_{\ell-1} = \mathbf{x} - \sum_{j=1}^{\ell-1} \hat{\mathbf{x}}_j. \tag{2.8}$$

Given the initial conditions $\tilde{\mathbf{x}}_0 = \mathbf{0}$ and $\mathbf{e}_0 = \mathbf{x}$, these vectors may be updated recursively according to the relations

$$\tilde{\mathbf{x}}_\ell = \tilde{\mathbf{x}}_{\ell-1} + \hat{\mathbf{x}}_\ell$$

$$\mathbf{e}_\ell = \mathbf{e}_{\ell-1} - \hat{\mathbf{x}}_\ell, \tag{2.9}$$

for $\ell \geq 1$. The goal of iterative vector approximation is then to determine the parameters of $\hat{\mathbf{x}}_\ell$ by minimizing the squared *sequential error norm* E_ℓ, defined as

$$\begin{aligned} E_\ell &= \|\mathbf{e}_\ell\|^2 \\ &= \|\mathbf{e}_{\ell-1} - \hat{\mathbf{x}}_\ell\|^2 \\ &= \|\mathbf{e}_{\ell-1} - \sum_{k=1}^{K} a_k^\ell \mathbf{v}_k^{i_\ell}\|^2, \end{aligned} \tag{2.10}$$

in terms of $\{a_1^\ell, \ldots, a_K^\ell\}$.

Assuming that the ensemble index i_ℓ is fixed, and assuming that the vectors $\{\mathbf{v}_1^{i_\ell}, \ldots, \mathbf{v}_K^{i_\ell}\}$ form a linearly independent set, this problem is simply a linear least-squares approximation of $\mathbf{e}_{\ell-1}$ by $\hat{\mathbf{x}}_\ell \in \mathcal{S}_{i_\ell}$, where $\mathcal{S}_{i_\ell} = \text{span}\{\mathbf{v}_1^{i_\ell}, \ldots, \mathbf{v}_K^{i_\ell}\}$. The *projection theorem* states that a necessary and sufficient condition for minimizing E_ℓ in terms of $\hat{\mathbf{x}}_\ell$ is that $\mathbf{e}_\ell$ be *orthogonal* to the subspace $\mathcal{S}_{i_\ell}$, or $\mathbf{e}_\ell \perp \mathcal{S}_{i_\ell}$.[1] Equivalently, this condition is met when $\mathbf{e}_\ell$ is orthogonal to the vectors making up $\hat{\mathbf{x}}_\ell$, i.e.

$$\langle \mathbf{e}_\ell, \mathbf{v}_m^{i_\ell} \rangle = 0 \quad 1 \leq m \leq K. \tag{2.11}$$

Substituting Equations 2.9 and 2.5 into this expression yields the *normal equations*,

$$\sum_{n=1}^{K} \gamma_{mn}^\ell a_n^\ell = \psi_m^\ell, \quad 1 \leq m \leq K, \tag{2.12}$$

where

$$\begin{aligned} \gamma_{mn}^\ell &= \langle \mathbf{v}_m^{i_\ell}, \mathbf{v}_n^{i_\ell} \rangle, \\ \psi_m^\ell &= \langle \mathbf{e}_{\ell-1}, \mathbf{v}_m^{i_\ell} \rangle. \end{aligned} \tag{2.13}$$

[1] As a result, the approximation $\hat{\mathbf{x}}_\ell$ is referred to as the *projection* of $\mathbf{e}_{\ell-1}$ onto $\mathcal{S}_{i_\ell}$

The normal equations may be written in matrix form as[2]

$$\begin{bmatrix} \gamma_{11}^{\ell} & \gamma_{12}^{\ell} & \cdots & \gamma_{1K}^{\ell} \\ \gamma_{12}^{\ell} & \gamma_{22}^{\ell} & \cdots & \gamma_{2K}^{\ell} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{1K}^{\ell} & \gamma_{2K}^{\ell} & \cdots & \gamma_{KK}^{\ell} \end{bmatrix} \begin{bmatrix} a_1^{\ell} \\ a_2^{\ell} \\ \vdots \\ a_K^{\ell} \end{bmatrix} = \begin{bmatrix} \psi_1^{\ell} \\ \psi_2^{\ell} \\ \vdots \\ \psi_K^{\ell} \end{bmatrix}, \tag{2.14}$$

or

$$\Gamma^{\ell}\mathbf{a}^{\ell} = \Psi^{\ell}. \tag{2.15}$$

The *Gram matrix* Γ^{ℓ} is symmetric, and since the vector set $\{\mathbf{v}_1^{i_\ell}, \ldots, \mathbf{v}_K^{i_\ell}\}$ is linearly independent, Γ^{ℓ} is also positive definite. Therefore, *Cholesky decomposition* [46] may be used to efficiently solve the normal equations for weighting coefficients which minimize E_ℓ.

At this point it is useful to derive an explicit expression for the minimum error term E_ℓ' in terms of the above results. Beginning with Equation 2.10 and substituting Equation 2.9, we have

$$\begin{aligned} E_\ell' &= \langle \mathbf{e}_\ell, \mathbf{e}_\ell \rangle \\ &= \langle \mathbf{e}_\ell, \mathbf{e}_{\ell-1} - \hat{\mathbf{x}}_\ell \rangle \\ &= \langle \mathbf{e}_\ell, \mathbf{e}_{\ell-1} \rangle - \langle \mathbf{e}_\ell, \hat{\mathbf{x}}_\ell \rangle. \end{aligned} \tag{2.16}$$

Since $\mathbf{e}_\ell \perp \mathcal{S}_{i_\ell}$, and since $\hat{\mathbf{x}}_\ell \in \mathcal{S}_{i_\ell}$, $\langle \mathbf{e}_\ell, \hat{\mathbf{x}}_\ell \rangle = 0$, thus

$$\begin{aligned} E_\ell' &= \langle \mathbf{e}_\ell, \mathbf{e}_{\ell-1} \rangle \\ &= \langle \mathbf{e}_{\ell-1}, \mathbf{e}_{\ell-1} \rangle - \langle \mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell \rangle \\ &= E_{\ell-1}' - \sum_{k=1}^{K} a_k^{\ell} \psi_k^{\ell}. \end{aligned} \tag{2.17}$$

This expression provides a means for computing the squared sequential error norm as an update of the previous error, given the optimal weighting coefficients $\{a_1^{\ell}, \ldots, a_K^{\ell}\}$ and the inner products $\{\psi_1^{\ell}, \ldots, \psi_K^{\ell}\}$ used to compute them.

[2] According to Equation 2.2, $\gamma_{mn}^{\ell} = \gamma_{nm}^{\ell}$

Solving the above normal equations determines optimal weighting coefficients for a single component of $\tilde{\mathbf{x}}$ assuming a fixed ensemble index value, but does not address the problem of determining an appropriate value of this last parameter. To this end an *ensemble search procedure* may be used. The simplest such procedure is an *exhaustive search*, whereby optimal weighting coefficients are calculated for each possible value of the ensemble index, yielding a corresponding value of E'_ℓ according to Equation 2.17. The optimal ensemble index i_ℓ is then chosen as that index value corresponding to the minimum error norm produced, and the weighting coefficients associated with this index value are used to construct the ℓ-th component $\hat{\mathbf{x}}_\ell$. Having determined parameters for $\hat{\mathbf{x}}_\ell$, the sequential approximation and error vectors are updated by Equation 2.9, and the procedure is repeated for the next component.

2.2 Properties of Iterative Vector Approximation

Several properties of the iterative vector approximation procedure described above provide insight into its operation and will later be useful for incorporating the technique in a sinusoidal modeling framework. The first and most important of these properties is seen by reconsidering the error update expression of Equation 2.17. Substituting Equation 2.9 into this expression and recalling that $\mathbf{e}_\ell$ is orthogonal to $\hat{\mathbf{x}}_\ell$ yields

$$\begin{aligned} E'_\ell &= E'_{\ell-1} - \langle \mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell \rangle \\ &= E'_{\ell-1} - \langle \mathbf{e}_\ell, \hat{\mathbf{x}}_\ell \rangle - \|\hat{\mathbf{x}}_\ell\|^2 \\ &= E'_{\ell-1} - \|\hat{\mathbf{x}}_\ell\|^2. \end{aligned} \tag{2.18}$$

This result is simply a generalized version of the well-known *Pythagorean theorem*. Since $\|\hat{\mathbf{x}}_\ell\| \geq 0$ for all $\hat{\mathbf{x}}_\ell \in \mathcal{S}_{i_\ell}$, the approximation error resulting from the addition of the ℓ-th component is therefore less than or equal to the previous error; hence, under no circumstances does iterative vector approximation result in increasing approximation error.

As will be shown next, under a mild condition on the vector ensembles, iterative vector approximation in fact yields only decreasing error:

Theorem 2.1 *If the union of the K vector ensembles spans $\boldsymbol{R}^p$, and $\mathbf{e}_{\ell-1} \neq \mathbf{0}$, then iterative vector approximation as described in Section 2.1 yields an optimal sequential error norm E_ℓ' with the property that $E_\ell' < E_{\ell-1}'$ for $\ell \geq 1$.*

Proof. Suppose that the union of the K vector ensembles spans $\boldsymbol{R}^p$ and that $\mathbf{e}_{\ell-1} \neq \mathbf{0}$, but that $E_\ell' = E_{\ell-1}'$. Then by Equation 2.18, $\|\hat{\mathbf{x}}_\ell\| = 0$ and therefore $\hat{\mathbf{x}}_\ell = \mathbf{0}$. Since the vector set $\{\mathbf{v}_1^i, \ldots, \mathbf{v}_K^i\}$ is linearly independent for any value of i, this implies that $a_k^i = 0$ for $1 \leq k \leq K$. Then, according to Equation 2.12, $\langle \mathbf{e}_{\ell-1}, \mathbf{v}_k^i \rangle = 0$ for $1 \leq k \leq K$ and for all i. Since the union of vector ensembles spans $\boldsymbol{R}^p$, this implies that $\mathbf{e}_{\ell-1} \perp \boldsymbol{R}^p$, thus $\mathbf{e}_{\ell-1} = \mathbf{0}$ in contradiction to the original premise. ∎

Under the condition that the entire space $\boldsymbol{R}^p$ is "covered" by the vector ensembles, this result guarantees that $\tilde{\mathbf{x}}_\ell$ converges to $\mathbf{x}$ as more components are added, providing theoretical justification for iterative vector approximation as well as an important guarantee of performance.

Another important property of iterative vector approximation relates the technique to simultaneous least-squares approximation in special circumstances:

Theorem 2.2 *If $JK \leq p$, and if the subspaces $\boldsymbol{S}_{i_1}, \ldots, \boldsymbol{S}_{i_J}$ are mutually orthogonal, then iterative vector approximation is equivalent to simultaneous least-squares approximation of $\mathbf{x}$ using the set of vectors denoted by*

$$\mathcal{V} = \bigcup_{j=1}^{J} \bigcup_{k=1}^{K} \mathbf{v}_k^{i_j}.$$

Proof. Referring to Equation 2.14, the normal equations associated with each component $\hat{\mathbf{x}}_\ell$ may be arranged in block-matrix form as

$$\begin{bmatrix} \Gamma^1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \Gamma^2 & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \Gamma^J \end{bmatrix} \begin{bmatrix} \mathbf{a}^1 \\ \mathbf{a}^2 \\ \vdots \\ \mathbf{a}^J \end{bmatrix} = \begin{bmatrix} \Psi^1 \\ \Psi^2 \\ \vdots \\ \Psi^J \end{bmatrix}.$$

If it is true that $\mathcal{S}_{i_m} \perp \mathcal{S}_{i_n}$ for $m \neq n$, then examining Equation 2.14 indicates that the left-hand block matrix in this equation corresponds to the Gram matrix of simultaneous least-squares approximation of $\mathbf{x}$ using the vector set $\mathcal{V}$ defined above. The column vector on the right-hand side of this matrix equation may be derived using Equations 2.13 and 2.8:

$$\begin{aligned} \psi_m^\ell &= \langle \mathbf{e}_{\ell-1}, \mathbf{v}_m^{i_\ell} \rangle \\ &= \langle \mathbf{x}, \mathbf{v}_m^{i_\ell} \rangle - \sum_{j=1}^{\ell-1} \langle \hat{\mathbf{x}}_j, \mathbf{v}_m^{i_\ell} \rangle. \end{aligned}$$

Since $\hat{\mathbf{x}}_j \in \mathcal{S}_{i_j}$, $\hat{\mathbf{x}}_j \perp \mathbf{v}_m^{i_\ell}$ for $j < \ell$ and therefore

$$\psi_m^\ell = \langle \mathbf{x}, \mathbf{v}_m^{i_\ell} \rangle.$$

The right-hand column vector thus corresponds to that of the normal equations in least-squares approximation of $\mathbf{x}$ using $\mathcal{V}$, and since $JK \leq p$ the above matrix equation possesses a unique solution identical to that produced by iterative vector approximation. ∎

This theorem is simply a block-matrix extension of the well-known result that a set of mutually orthogonal vectors produces a diagonal Gram matrix.

As is clear from the above theorem, given a set of ensemble indices $\{i_\ell\}$, iterative vector approximation achieves optimal performance only when the resulting component vectors $\{\hat{\mathbf{x}}_\ell\}$ are mutually orthogonal or *uncorrelated*. In order to quantify the effect of correlation between components on the performance of iterative vector approximation, it is useful to define a measure of correlation between a vector and a subspace. Referring to Equation 2.18, we have the result that $\langle \mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell \rangle = \|\hat{\mathbf{x}}_\ell\|^2$; thus, the correlation coefficient between $\mathbf{e}_{\ell-1}$ and $\hat{\mathbf{x}}_\ell$ may be written as

$$\rho(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell) = \frac{\langle \mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell \rangle}{\|\mathbf{e}_{\ell-1}\| \|\hat{\mathbf{x}}_\ell\|} = \frac{\|\hat{\mathbf{x}}_\ell\|^2}{\|\mathbf{e}_{\ell-1}\| \|\hat{\mathbf{x}}_\ell\|} = \frac{\|\hat{\mathbf{x}}_\ell\|}{\|\mathbf{e}_{\ell-1}\|}. \tag{2.19}$$

This correlation coefficient has the property that $0 \leq \rho(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell) \leq 1$; as a result, the concept of an angle between vectors [47] may be introduced by setting

$$\cos[\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell)] = \rho(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell).$$

The vector angle $\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell)$ has the property that $0 \leq \theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell) \leq \pi/2$, with $\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell) = 0$ indicating collinearity and $\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell) = \pi/2$ indicating orthogonality. It is useful to note that E'_ℓ may be expressed in terms of $\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell)$ as

$$E'_\ell = E'_{\ell-1}\{1 - \cos^2[\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell)]\}. \tag{2.20}$$

Since the least-squares approximation $\hat{\mathbf{x}}_\ell$ is unique for a given $\mathbf{e}_{\ell-1}$ and subspace $\boldsymbol{S}_{i_\ell}$, the angle $\theta(\mathbf{e}_{\ell-1}, \hat{\mathbf{x}}_\ell)$ is easily shown to be equivalent to the *principal angle* between $\boldsymbol{S}_{i_\ell}$ and the subspace spanned by $\hat{\mathbf{x}}_\ell$ [48]. This observation is generalized in the following definition:

Definition 2.1 *Given a vector* $\mathbf{x} \in \boldsymbol{R}^p$ *and a subspace* $\boldsymbol{Y}$, *the "principal angle" between* $\mathbf{x}$ *and* $\boldsymbol{Y}$ *is defined by*

$$\cos[\theta(\mathbf{x}, \boldsymbol{Y})] = \cos[\theta(\mathbf{x}, \tilde{\mathbf{x}}_Y)] = \frac{\|\mathbf{x}_Y\|}{\|\mathbf{x}\|},$$

where $\mathbf{x}_Y$ *is the projection of* $\mathbf{x}$ *onto* $\boldsymbol{Y}$.

The principal angle $\theta(\mathbf{x}, \boldsymbol{Y})$ may be interpreted as a measure of how "close" $\mathbf{x}$ is to the subspace $\boldsymbol{Y}$. For instance, if $\mathbf{x} \in \boldsymbol{Y}$, then $\theta(\mathbf{x}, \boldsymbol{Y}) = 0$; likewise, if $\mathbf{x} \perp \boldsymbol{Y}$, then $\theta(\mathbf{x}, \boldsymbol{Y}) = \pi/2$. In general, smaller values of $\theta(\mathbf{x}, \boldsymbol{Y})$ indicate greater correlation between $\mathbf{x}$ and $\boldsymbol{Y}$.

We are now prepared to quantify the performance of iterative vector approximation in terms of $\theta(\mathbf{x}, \boldsymbol{Y})$:

Theorem 2.3 *The sequential error norm* E'_ℓ *has the following property:*

$$E'_\ell \geq \cos^2[\theta(\hat{\mathbf{x}}_\ell, \boldsymbol{S}_{i_{\ell-1}})]E'_{\ell-1}.$$

Proof. The component vector $\hat{\mathbf{x}}_\ell$ may be expressed as its projection onto the subspace $\boldsymbol{S}_{i_{\ell-1}}$ plus an error vector $\mathbf{e}_x$, hence

$$\mathbf{e}_x = \hat{\mathbf{x}}_\ell - \mathbf{s}_{\ell-1},$$

where $\mathbf{s}_{\ell-1} \in \boldsymbol{S}_{i_{\ell-1}}$. According to Equation 2.20, the squared norm of this error vector is given by

$$\|\mathbf{e}_x\|^2 = \|\hat{\mathbf{x}}_\ell\|^2\{1 - \cos^2[\theta(\hat{\mathbf{x}}_\ell, \boldsymbol{S}_{i_{\ell-1}})]\}.$$

Likewise, the error vector $\mathbf{e}_x$ is given in terms of its least-squares approximation by $\mathbf{e}_{\ell-1}$ as

$$\mathbf{e}_x = \frac{\langle \mathbf{e}_x, \mathbf{e}_{\ell-1}\rangle}{\|\mathbf{e}_{\ell-1}\|^2}\mathbf{e}_{\ell-1} + \mathbf{e}_{\ell-1}^{\perp},$$

where

$$\mathbf{e}_{\ell-1}^{\perp} = \mathbf{e}_x - \frac{\langle \mathbf{e}_x, \mathbf{e}_{\ell-1}\rangle}{\|\mathbf{e}_{\ell-1}\|^2}\mathbf{e}_{\ell-1} \perp \mathbf{e}_{\ell-1}.$$

Recalling that $\mathbf{e}_{\ell-1} \perp \boldsymbol{S}_{\ell-1}$ and referring to Equation 2.18, the inner product term in this expression is given by

$$\langle \mathbf{e}_x, \mathbf{e}_{\ell-1}\rangle = \langle \hat{\mathbf{x}}_\ell - \mathbf{s}_{\ell-1}, \mathbf{e}_{\ell-1}\rangle = \langle \hat{\mathbf{x}}_\ell, \mathbf{e}_{\ell-1}\rangle = \|\hat{\mathbf{x}}_\ell\|^2.$$

Using the second expression for $\mathbf{e}_x$, the squared norm of $\mathbf{e}_x$ is thus expressed as

$$\|\mathbf{e}_x\|^2 = \frac{\|\hat{\mathbf{x}}_\ell\|^4}{\|\mathbf{e}_{\ell-1}\|^2} + \|\mathbf{e}_{\ell-1}^{\perp}\|^2.$$

Substituting the first expression for $\|\mathbf{e}_x\|^2$ into this equation and noting that $\|\mathbf{e}_{\ell-1}^{\perp}\|^2 \geq 0$ yields

$$\frac{\|\hat{\mathbf{x}}_\ell\|^2}{E'_{\ell-1}} \leq \{1 - \cos^2\theta[(\hat{\mathbf{x}}_\ell, \boldsymbol{S}_{i_{\ell-1}})]\}.$$

Rearranging terms on both sides of this inequality leads to

$$E'_\ell = E'_{\ell-1} - \|\hat{\mathbf{x}}_\ell\|^2 \geq \cos^2[\theta(\hat{\mathbf{x}}_\ell, \boldsymbol{S}_{i_{\ell-1}})]E'_{\ell-1}.$$

■

The result of this theorem is a lower bound on the approximation error E'_ℓ in terms of the amount of correlation between the component vector $\hat{\mathbf{x}}_\ell$ and the space spanned by $\{\mathbf{v}_1^{i_{\ell-1}}, \ldots, \mathbf{v}_K^{i_{\ell-1}}\}$. Theorem 2.2 indicates that mutually orthogonal component vectors perform optimally in iterative vector approximation; conversely, Theorem 2.3 shows that highly correlated component vectors result in inefficient approximation, and that performance improves as the components become less correlated.

It is important to note that Theorem 2.3 only addresses the question of correlation from one component to the next: although no simple proof exists to establish the result, experience indicates that if a given component vector $\hat{\mathbf{x}}_\ell$ is highly correlated with *any* previous component, approximation will be inefficient.

Another useful property of iterative vector approximation is determined by examining Equation 2.18. Since E'_ℓ is the square of a norm, it must be true that $E'_\ell \geq 0$. This, combined with the result from Theorem 2.1, leads to the expression

$$0 \leq E'_{\ell-1} - \|\hat{\mathbf{x}}_\ell\|^2 < E'_{\ell-1}.$$

When rearranged, this becomes

$$0 < \|\hat{\mathbf{x}}_\ell\|^2 \leq E'_{\ell-1} \tag{2.21}$$

for $\ell \geq 1$. In other words, the squared norm of successive components is nonzero, positive, and bounded above by a quantity which decreases with increasing ℓ. If the component vectors are mutually orthogonal as in Theorem 2.2, then the component norms will be a strictly nonincreasing sequence, since in this case the norm of any given component is independent of other components. Although in the case where components are correlated it can be shown by counterexample that $\|\hat{\mathbf{x}}_\ell\|^2$ need not decrease with increasing ℓ, it has been observed that the closer component vectors are to mutual orthogonality, the more the component vector norms follow a decreasing trend.

CHAPTER 3

Analysis-by-Synthesis/Overlap-Add (ABS/OLA) Sinusoidal Model

Given an understanding of the iterative vector approximation procedure described in Chapter 2, the question which remains is how to apply this approximation technique in a useful way to the problem of sinusoidal signal modeling. This chapter describes in detail a sinusoidal model formulation whose parameters may be determined using an analysis-by-synthesis technique, and discusses how analysis-by-synthesis may be combined with iterative vector approximation. A good deal of signal notation is introduced in this chapter, and is listed in order of appearance at the end of the chapter. Henceforth, a discrete-time sequence $s[n]$ is assumed to be a sampled version of a continuous-time audio signal $s_c(t)$, sampled at a rate of F_s samples/sec, *i.e.*

$$s[n] = s_c(n/F_s), \tag{3.1}$$

where $s_c(t)$ is assumed bandlimited to $F_s/2$ Hz.

3.1 Synthesis Model

The model proposed to represent $s[n]$ is an overlap-add sinusoidal model formulation given in its most general form by

$$\tilde{s}[n] = \sigma[n] \sum_{k=-\infty}^{\infty} w_s[n - kN_s]\tilde{s}^k[n - kN_s]. \tag{3.2}$$

The *synthesis window* $w_s[n]$ is a complementary window obeying the constraint

$$\sum_{k=-\infty}^{\infty} w_s[n - kN_s] = 1, \tag{3.3}$$

for all n, where N_s determines synthesis frame length as discussed in Section 1.3.1. The k-th *synthetic contribution*, $\tilde{s}^k[n]$, is given by

$$\begin{aligned} \tilde{s}^k[n] &= \sum_{j=1}^{J[k]} A_j^k \cos(2\pi f_j^k n/F_s + \phi_j^k) \\ &= \sum_{j=1}^{J[k]} A_j^k \cos(\omega_j^k n + \phi_j^k), \end{aligned} \tag{3.4}$$

where $0 \leq f_j^k \leq F_s/2$, and the *envelope sequence* $\sigma[n]$ reflects time-varying changes in the energy of $s[n]$; its purpose in the model is to boost accuracy during transitory regions of $s[n]$. In words, $\tilde{s}[n]$ is a sum of window-weighted synthetic waveforms overlapped by N_s samples, added together and modulated by $\sigma[n]$, where each synthetic waveform is produced by adding together sinusoids of various amplitudes, frequencies and phases.

This sinusoidal model formulation resembles overlap-add synthesis using the DSTFT in that constant-amplitude, constant-frequency sinusoids (derived from complex exponentials) are used to represent $s[n]$ on a frame-by-frame basis, but differs in its use of a modulating envelope sequence $\sigma[n]$, variable numbers of sinusoidal components, and arbitrary component frequencies. Note that when $\sigma[n] \equiv 1$ the model formulation is identical to that defined in Equations 1.10 and 1.11 as discussed in Section 1.3.1. While any complementary window $w_s[n]$ will suffice, a symmetric, tapered window such as a triangular window or a Hanning window of the form

$$w_s[n] = \begin{cases} \cos^2(n\pi/2N_s), & |n| \leq N_s \\ 0, & \text{otherwise,} \end{cases} \tag{3.5}$$

is typically used. With a synthesis window of length $2N_s + 1$, a synthesis frame of N_s samples of $\tilde{s}[n]$ may be expressed in a relatively compact form:

$$\tilde{s}[n + kN_s] = \sigma[n + kN_s](w_s[n]\tilde{s}^k[n] + w_s[n - N_s]\tilde{s}^{k+1}[n - N_s]), \tag{3.6}$$

for $0 \leq n < N_s$. Figure 3.1 illustrates a synthesis frame and the overlapping synthetic contributions which produce it. As with any frame-based approach to speech modeling, care must be taken in choosing N_s such that the speech signal may be assumed stationary over a given frame interval. Typical values of N_s correspond to between 5 and 20 msec, depending on application requirements. The parameter set which must

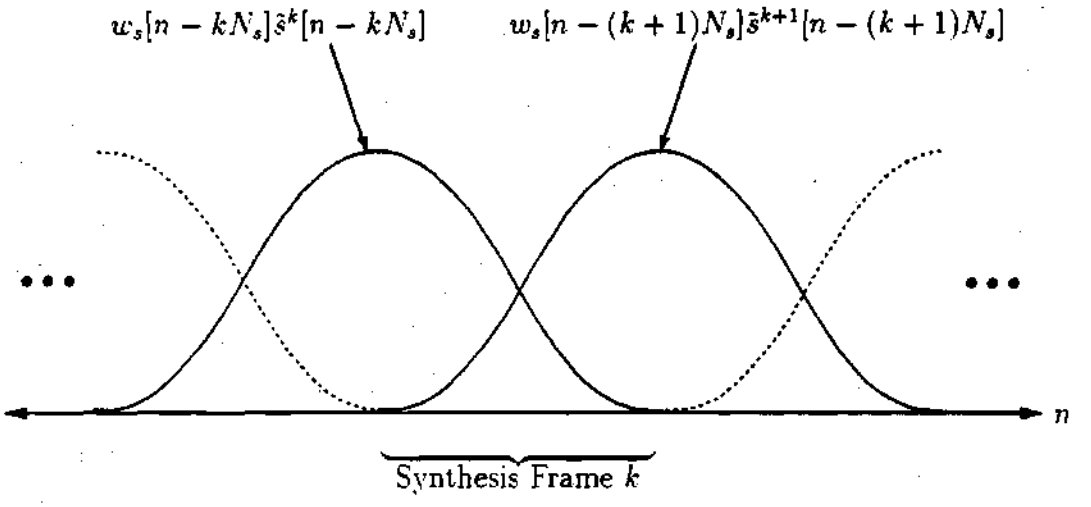

Figure 3.1: Illustration of overlap-add synthesis structure using complementary synthesis windows.

be determined in order to represent a given $s[n]$ consists of the envelope sequence $\sigma[n]$ and the amplitudes $\{A_j^k\}$, frequencies $\{\omega_j^k\}$ and phases $\{\phi_j^k\}$ of each synthetic contribution $\tilde{s}^k[n]$. The problem of determining $\sigma[n]$ is discussed first.

3.2 Envelope Estimation

The envelope sequence $\sigma[n]$, which reflects syllabic variations in the average magnitude of $s[n]$, can be reasonably estimated by lowpass filtering $|s[n]|$ [49]. While a wide variety of lowpass *envelope estimation filters* may be used, a recursive implementation is desirable since $\sigma[n]$ is required for all n. Early experimentation was with a simple

first-order recursive filter defined by the difference equation

$$\sigma[n] = \lambda\sigma[n-1] + (1-\lambda)|s[n]|, \tag{3.7}$$

where $0 < \lambda < 1$. While this filter has the advantage of simplicity, requiring two multiplies and one add per sample to compute, it exhibits poor performance as an envelope estimation filter. Large values of λ are required to make the envelope estimation filter sufficiently lowpass to eliminate harmonic components from $\sigma[n]$, but large values of λ imply slow response to energy changes, defeating the purpose of the envelope sequence. Figure 3.2(a) demonstrates a large amount of ripple resulting from a small value of λ, and 3.2(b) shows how, for a larger value of λ, $\sigma[n]$ tracks temporal energy variations poorly and still exhibits some ripple.

It was noted, however, that given $\sigma[n]$ from Equation 3.7 with a smaller value of λ, the resulting ripple could be reduced by applying the same filter to $\sigma[n]$, without losing the trends in syllabic energy. Repeating this process I times results in an envelope estimation filter defined by the recursive relation

$$y_i[n] = \lambda y_i[n-1] + (1-\lambda)y_{i-1}[n], \quad 1 \le i \le I, \quad n \ge 0, \tag{3.8}$$

where $y_0[n] = |s[n]|$ and where $y_i[-1] = 0$ for $1 \le i \le I$. In this formulation, the envelope sequence is then given by

$$\sigma[n] = y_I[n + n_\sigma]. \tag{3.9}$$

Since the envelope estimation filter is causal, the shift of n_σ samples is necessary to account for the delay introduced by filtering, ensuring temporal correlation between $s[n]$ and $\sigma[n]$.

The envelope estimation filter defined in Equation 3.8 has the transfer function

$$F(e^{j\omega}) = \left(\frac{1-\lambda}{1-\lambda e^{-j\omega}}\right)^I. \tag{3.10}$$

The nature of this filter may be easily understood by applying statistical principles. According to Equation 3.10, this filter may be viewed as the cascade of I filters, each

with the causal impulse response

$$h[n] = (1 - \lambda)\lambda^n u[n].$$

The impulse response of the overall filter, $f[n]$, is then simply the convolution of $h[n]$ with itself I times. Since $h[n]$ takes only positive values, and since

$$F(e^{j0}) = 1 = \sum_{n=0}^{\infty} f[n],$$

the *central limit theorem* [50] states that with increasing I, $f[n]$ tends toward a sampled Gaussian curve, *i.e.*

$$f[n] \approx \frac{1}{\sigma_f\sqrt{2\pi}} e^{-(n-\mu_f)^2/2\sigma_f^2}, \tag{3.11}$$

where the *mean*, μ_f, corresponds to the envelope estimation filter delay n_σ, and where σ_f, the *standard deviation*, determines the frequency selectivity of the filter.

Since, according to this interpretation, $f[n]$ is viewed as a discrete *probability density function*, the value of n_σ may be calculated as

$$n_\sigma \approx \sum_{n=0}^{\infty} n f[n] = j\frac{d}{d\omega} F(e^{j\omega})\bigg|_{\omega=0}. \tag{3.12}$$

By substituting the expression for $F(e^{j\omega})$ given in Equation 3.10, n_σ is approximately given by[1]

$$n_\sigma = \left\langle I\frac{\lambda}{1-\lambda} \right\rangle. \tag{3.13}$$

While it is possible to calculate σ_f given in Equation 3.11 in terms of I and λ, the calculation is rather involved and not necessary. Instead, empirical testing on speech waveforms sampled at $F_s = 8000$ samples/sec has demonstrated that values of $\lambda = .9$, $I = 20$ and $n_\sigma = 180$ at this sampling rate yield satisfactory performance over speakers with various average pitch frequencies. In order to maintain this performance over a range of sampling rates, λ (and n_σ) must be varied to maintain similar frequency

[1] where $\langle \cdot \rangle$ represents the "round to nearest integer" operator.

selectivity. A simple method of achieving this goal is to consider the filter response $h[n]$ to be sampled from the continuous response $h_c(t) = (1 - \lambda_c)\lambda_c^t u(t)$, and to vary λ to maintain a fixed 3 dB filter bandwidth. This leads to the relation

$$\lambda(F_s) = .9^{F_s/8000}. \tag{3.14}$$

Figure 3.2(c) demonstrates the effect of quasi-Gaussian filtering applied to the estimation of $\sigma[n]$. Compared to first-order filters, the envelope sequence resulting from this approach is seen to closely follow the desired time-varying trends in signal energy, while eliminating undesirable ripple effects. The cost of quasi-Gaussian filtering is, of course, a considerably increased computational load over the simpler filters.

3.3 Analysis-by-Synthesis

Given $\sigma[n]$, the objective of analysis is to determine amplitude, frequency and phase parameters for each $\tilde{s}^k[n]$ in Equation 3.2 such that $\tilde{s}[n]$ is "closest" to $s[n]$ in some sense. An approach typically employed to solve problems of this type is to minimize the mean-square error

$$E = \sum_{n=-\infty}^{\infty} \{s[n] - \tilde{s}[n]\}^2 \tag{3.15}$$

in terms of the parameters of $\tilde{s}[n]$. However, attempting to solve this problem simultaneously for all the parameters is not practical.

Fortunately, if $s[n]$ is approximately stationary over short time intervals, it is feasible to solve for the amplitude, frequency and phase parameters of $\tilde{s}^k[n]$ in isolation by approximating $s[n]$ over an *analysis frame* of length $2N_a + 1$ samples centered at $n = kN_s$. The synthetic contribution $\tilde{s}^k[n]$ may then be determined by minimizing

$$E^k = \sum_{n=-N_a}^{N_a} w_a[n]\{s[n + kN_s] - \sigma[n + kN_s]\tilde{s}^k[n]\}^2 \tag{3.16}$$

with respect to the amplitudes, frequencies and phases of $\tilde{s}^k[n]$. The *analysis window* $w_a[n]$ may be an arbitrary positive function, but is typically a symmetric, tapered

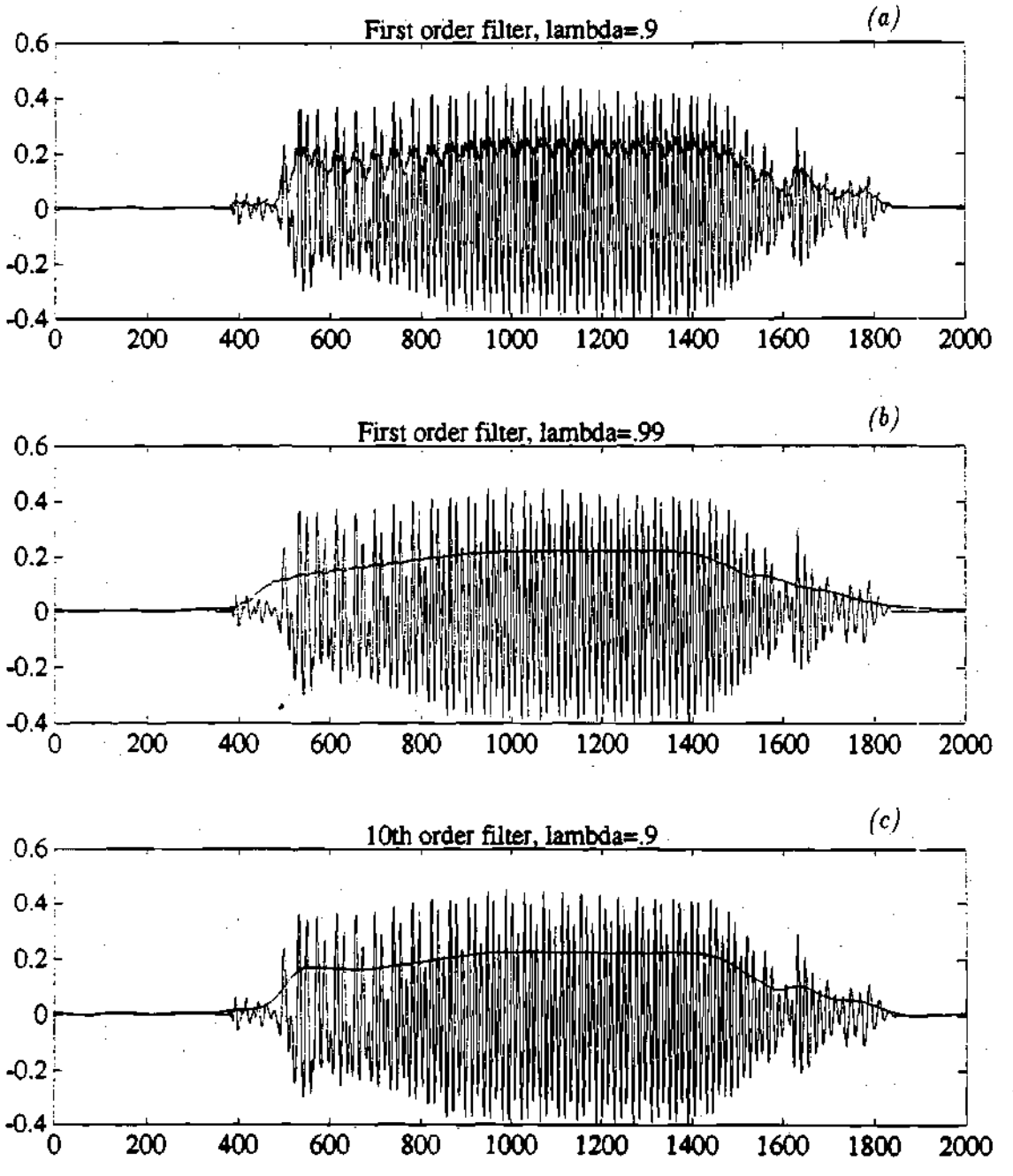

Figure 3.2: Plots of a speech segment and envelope sequence determined by *(a)* first-order recursive filter (λ=.9), *(b)* first-order recursive filter (λ=.99), and *(c)* quasi-Gaussian lowpass filter.

window which serves to force greater accuracy at the frame center, where the contribution of $\tilde{s}^k[n]$ to $\tilde{s}[n]$ is dominant. Strategies for choosing an analysis window function and appropriate values of N_a will be discussed later, but in order to ensure the accuracy of $\tilde{s}[n]$, N_a should be greater than or equal to N_s.

Defining $x[n]$ and $g[n]$ by

$$\begin{aligned} x[n] &\triangleq (w_a[n])^{1/2} s[n + kN_s] \\ g[n] &\triangleq (w_a[n])^{1/2} \sigma[n + kN_s], \end{aligned} \tag{3.17}$$

and making use of Equation 3.4, E^k may be rewritten as

$$\begin{aligned} E &= \sum_{n=-N_a}^{N_a} \{x[n] - \tilde{x}[n]\}^2 \\ &= \sum_{n=-N_a}^{N_a} \{x[n] - \sum_{j=1}^{J} \hat{x}_j[n]\}^2 \\ &= \sum_{n=-N_a}^{N_a} \{x[n] - \sum_{j=1}^{J} A_j g[n] \cos(\omega_j n + \phi_j)\}^2, \end{aligned} \tag{3.18}$$

where the frame notation superscripts have been omitted to simplify the equations. Unfortunately, without *a priori* knowledge of the frequency parameters, this minimization problem is highly nonlinear and therefore very difficult to solve. Various techniques have been proposed to determine, in a closed-form fashion, an appropriate set of component frequencies to use in a sinusoidal representation. These approaches typically involve computationally intensive procedures such as eigenanalysis [51] or high-order polynomial rooting [52], and are often sensitive to the number of components assumed in the model, which is itself unknown beforehand.

As an alternative, a slightly suboptimal but relatively efficient analysis-by-synthesis algorithm may be employed to determine model parameters. Generally speaking, analysis-by-synthesis operates by varying the parameters of a given signal model in a systematic, iterative fashion to determine parameters which yield a minimum error measure obtainable within the constraints of the parameter variation procedure [53]. In the case of sinusoidal modeling, an effective and efficient analysis-

by-synthesis technique may be formulated by incorporating the iterative vector approximation technique described in Chapter 2. This was first suggested in [54] and refined in [55].

Comparing Equation 3.18 with Equations 2.6 and 2.3, the mean-square error term in this expression is seen to be identical to the approximation error norm defined in Equation 2.6 on the vector space $\boldsymbol{R}^{(2N_a+1)}$. In this interpretation, using the notation of Equation 2.1, the *signal vector* being approximated is given by

$$\mathbf{x} = (x[-N_a], x[-N_a+1], \ldots, x[N_a]). \tag{3.19}$$

Furthermore, the approximation to $\mathbf{x}$ is made up of J component vectors $\{\hat{\mathbf{x}}_j\}$ as in Equation 2.4, where

$$\begin{aligned} \hat{\mathbf{x}}_j &= (A_j g[-N_a]\cos(-\omega_j N_a + \phi_j), A_j g[-N_a+1]\cos(\omega_j(-N_a+1)+\phi_j), \\ &\qquad \ldots, A_j g[N_a]\cos(\omega_j N_a + \phi_j)). \end{aligned} \tag{3.20}$$

Note that each component sequence $\hat{x}_j[n]$ may be expressed as

$$\begin{aligned} \hat{x}_j[n] &= A_j g[n]\cos(\omega_j n + \phi_j) \\ &= a_1^j g[n]\cos\omega_j n + a_2^j g[n]\sin\omega_j n, \end{aligned} \tag{3.21}$$

where the *quadrature parameters* are given by

$$\begin{aligned} a_1^j &= A_j\cos\phi_j \\ a_2^j &= -A_j\sin\phi_j. \end{aligned} \tag{3.22}$$

By constraining the frequency parameters $\{\omega_j\}$ to be of the form

$$\omega_j = 2\pi i_j/M, \tag{3.23}$$

where $0 \le i_j \le M/2$ (assuming M is even), each component vector $\hat{\mathbf{x}}_j$ is clearly a linear combination of $K = 2$ ensemble vectors as in Equation 2.5, where

$$\begin{aligned} \mathbf{v}_1^{i_j} &= (g[-N_a]\cos\omega_j(-N_a), \ldots, g[N_a]\cos\omega_j N_a) \\ \mathbf{v}_2^{i_j} &= (g[-N_a]\sin\omega_j(-N_a), \ldots, g[N_a]\sin\omega_j N_a). \end{aligned} \tag{3.24}$$

The two vector ensembles, one corresponding to cosinusoidal sequences of various frequencies weighted by $g[n]$ and the other to weighted sinusoidal sequences, each have $I = M/2 + 1$ members corresponding to equally spaced frequencies in the range $[0, \pi]$.

Rewriting Equation 2.9 in terms of sequential approximation and sequential error sequences corresponding to signal vectors and in terms of sinusoidal model parameters yields

$$\begin{aligned} \hat{x}_\ell[n] &= \hat{x}_{\ell-1}[n] + \hat{x}_\ell[n] = \hat{x}_{\ell-1}[n] + A_\ell g[n]\cos(\omega_\ell n + \phi_\ell) \\ e_\ell[n] &= e_{\ell-1}[n] - \hat{x}_\ell[n] = e_{\ell-1}[n] - A_\ell g[n]\cos(\omega_\ell n + \phi_\ell) \end{aligned} \tag{3.25}$$

for $|n| \leq N_a$. At this point the iterative vector approximation algorithm described in Chapter 2 may be applied directly to the sinusoidal modeling problem. The resulting 2×2 set of normal equations is given by

$$\begin{aligned} \gamma_{11}^\ell a_1^\ell + \gamma_{12}^\ell a_2^\ell &= \psi_1^\ell \\ \gamma_{12}^\ell a_1^\ell + \gamma_{22}^\ell a_2^\ell &= \psi_2^\ell, \end{aligned} \tag{3.26}$$

where

$$\begin{aligned} \gamma_{11}^\ell &= \|\mathbf{v}_1^{i_\ell}\|^2 = \sum_{n=-N_a}^{N_a} g^2[n]\cos^2\omega_\ell n \\ \gamma_{12}^\ell &= \langle \mathbf{v}_1^{i_\ell}, \mathbf{v}_2^{i_\ell} \rangle = \sum_{n=-N_a}^{N_a} g^2[n]\cos\omega_\ell n \sin\omega_\ell n \\ \gamma_{22}^\ell &= \|\mathbf{v}_2^{i_\ell}\|^2 = \sum_{n=-N_a}^{N_a} g^2[n]\sin^2\omega_\ell n \\ \psi_1^\ell &= \langle \mathbf{e}_{\ell-1}, \mathbf{v}_1^{i_\ell} \rangle = \sum_{n=-N_a}^{N_a} e_{\ell-1}[n]g[n]\cos\omega_\ell n \\ \psi_2^\ell &= \langle \mathbf{e}_{\ell-1}, \mathbf{v}_2^{i_\ell} \rangle = \sum_{n=-N_a}^{N_a} e_{\ell-1}[n]g[n]\sin\omega_\ell n. \end{aligned} \tag{3.27}$$

These normal equations may be solved directly for a_1^ℓ and a_2^ℓ, yielding

$$\begin{aligned} a_1^\ell &= (\gamma_{22}^\ell \psi_1^\ell - \gamma_{12}^\ell \psi_2^\ell)/\Delta_\Gamma \\ a_2^\ell &= (\gamma_{11}^\ell \psi_2^\ell - \gamma_{12}^\ell \psi_1^\ell)/\Delta_\Gamma, \end{aligned} \tag{3.28}$$

where $\Delta_\Gamma = \gamma_{11}^\ell \gamma_{22}^\ell - (\gamma_{12}^\ell)^2$. By Equation 2.17, given a_1^ℓ and a_2^ℓ we can calculate E_ℓ by

$$E_\ell = E_{\ell-1} - a_1^\ell \psi_1^\ell - a_2^\ell \psi_2^\ell. \tag{3.29}$$

Having determined a_1^ℓ and a_2^ℓ, the amplitude and phase parameters of the ℓ-th component are given by the relations[2]

$$\begin{aligned} A_\ell &= [(a_1^\ell)^2 + (a_2^\ell)^2]^{1/2} \\ \phi_\ell &= -\tan^{-1}(a_2^\ell / a_1^\ell). \end{aligned} \tag{3.30}$$

A necessary and sufficient condition for the invertibility of the Gram matrix in this case is that the *Gram determinant* Δ_Γ be nonzero. Noting that the parameters of Equation 3.27 are inner products, the well-known *Cauchy-Schwartz inequality* states that

$$0 \leq \frac{\langle \mathbf{v}_1^{i_\ell}, \mathbf{v}_2^{i_\ell} \rangle^2}{\|\mathbf{v}_1^{i_\ell}\|^2 \|\mathbf{v}_2^{i_\ell}\|^2} = \frac{(\gamma_{12}^\ell)^2}{\gamma_{11}^\ell \gamma_{22}^\ell} \leq 1$$

and therefore

$$\Delta_\Gamma \geq 0;$$

furthermore, $\Delta_\Gamma = 0$ only when $\mathbf{v}_1^{i_\ell}$ is some constant multiple of $\mathbf{v}_2^{i_\ell}$, or vice-versa. Referring to Equation 3.24 and assuming that $g[n] > 0$, since $\cos(0) = 1$ and $\sin(0) = 0$, then $\mathbf{v}_1^{i_\ell}$ cannot be a multiple of $\mathbf{v}_2^{i_\ell}$, and the only circumstance where the reverse is true is when $\mathbf{v}_2^{i_\ell} = \mathbf{0}$, which only occurs when $\omega_\ell = 0$ or $\omega_\ell = \pi$. Therefore, if $g[n] > 0$ the Gram matrix is invertible for all frequencies except $\omega_\ell = 0$ or $\omega_\ell = \pi$. At the degenerate frequencies a unique solution is forced by assuming that $b_1^\ell = 0$ and solving the least-squares problem in terms of $\mathbf{v}_1^{i_\ell}$ alone, yielding the solution

$$a_1^\ell = \psi_1^\ell / \gamma_{11}^\ell \tag{3.31}$$

As discussed in Chapter 2, an appropriate ensemble index value for each component may be determined using an exhaustive ensemble search procedure. Referring to

[2]These relations are verified by substituting the quadrature parameters defined in Equation 3.22.

Equation 3.23, this is equivalent to a frequency search over the candidate frequency set given by $\omega_c[i] = 2i\pi/M$ for $0 \leq i \leq M/2$. For each $\omega_c[i]$, the corresponding value of E_ℓ is calculated using Equation 3.29, and ω_ℓ is chosen as that value of $\omega_c[i]$ which yields the minimum error. A_ℓ and ϕ_ℓ are chosen as the amplitude and phase parameters associated with that frequency value, and the procedure is repeated to determine the next sinusoidal component. Figure 3.3 shows a functional block diagram of the analysis procedure just described, illustrating its iterative, "closed-loop" structure. What may be noted from this figure is that the parameters of each successive component are chosen to minimize the error "left over" after approximation by previous components. Figure 3.4 illustrates this characteristic of analysis-by-synthesis using an example of analysis-by-synthesis applied to a segment of speech.

An important consideration in the analysis-by-synthesis algorithm proposed is whether $\tilde{x}_\ell[n]$ converges to $x[n]$. As discussed in Theorem 2.1, convergence is guaranteed provided the union of vector ensembles spans the space $\boldsymbol{R}^{(2N_a+1)}$. Referring to Equation 3.24, the ensemble vector elements are expressed in functional form as

$$\begin{aligned} v_1^i[n] &= g[n]\cos 2\pi in/M = g[n](e^{j2\pi in/M} + e^{-j2\pi in/M})/2 \\ v_2^i[n] &= g[n]\sin 2\pi in/M = g[n](e^{j2\pi in/M} - e^{-j2\pi in/M})/2j. \end{aligned} \tag{3.32}$$

Since the ensemble vectors are themselves constrained combinations of complex exponential vectors, if the sequence $g[n]$ is nonzero then the space spanned by the union of ensembles is a subspace of the space spanned by the complex basis vectors $\{\hat{\mathbf{v}}^i\}$ whose elements are given by[3]

$$\hat{v}^i[n] = e^{j2\pi in/M}, \quad 0 \leq i \leq M-1. \tag{3.33}$$

As is well known, these vectors are mutually orthogonal in the complex space $\boldsymbol{C}^M$ of complex "M-tuples," therefore the basis vectors $\{\hat{\mathbf{v}}^i\}$ span $\boldsymbol{C}^M$. Clearly, $\boldsymbol{R}^{(2N_a+1)}$ is a subspace of $\boldsymbol{C}^M$ if and only if $M > 2N_a$. Furthermore, if $g[n] = 0$

[3]Note that $e^{-j2\pi in/M} = e^{j2\pi(M-i)n/M}$.

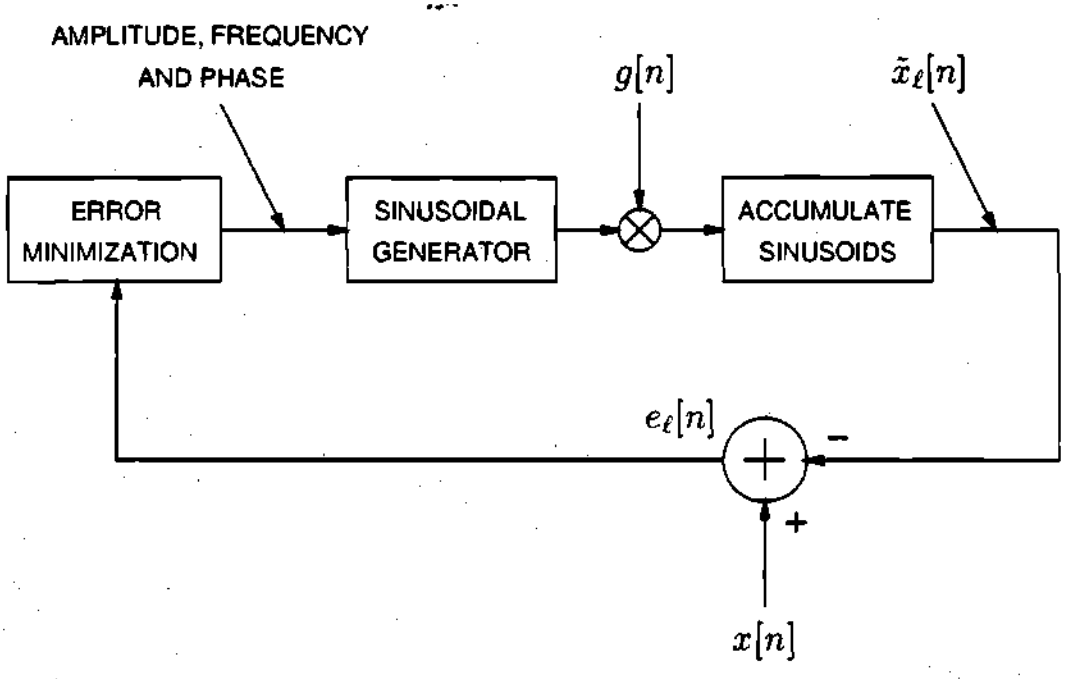

Figure 3.3: Block diagram of analysis-by-synthesis procedure applied to overlap-add sinusoidal modeling.

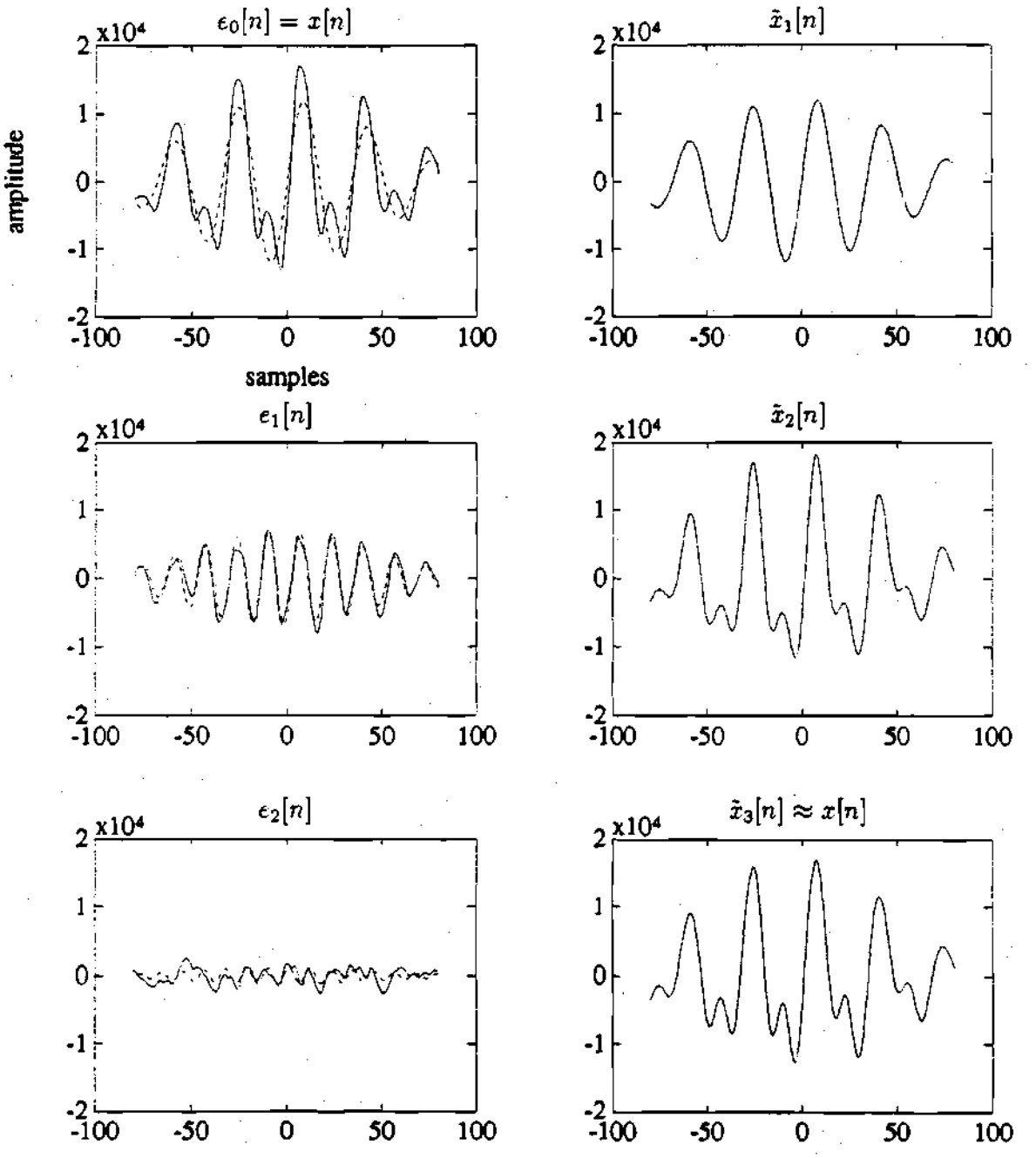

Figure 3.4: Illustration of analysis-by-synthesis applied to sinusoidal modeling. Left-hand plots show sequential error sequences $e_\ell[n]$, with best approximations $\hat{x}_\ell[n]$ dotted. Right-hand plots show sequential approximations $\tilde{x}_\ell[n]$.

for any $n \in [-N_a, N_a]$, then the vector ensembles cannot span $\boldsymbol{R}^{(2N_a+1)}$. Therefore the conditions $g[n] \neq 0$ and $M > 2N_a$ are sufficient to guarantee convergence in analysis-by-synthesis.

As a practical matter, the value of M used in analysis-by-synthesis will be considerably larger than $2N_a$, in order to provide a fine grid of candidate frequencies so that component frequencies are well estimated. Of course, larger values of M imply more analysis computation, so the choice of M is a tradeoff. In order to provide a level of accuracy that is independent of the analysis frame length, M should be proportional to N_a, *i.e.*

$$M = \nu_a N_a$$

where ν_a is typically in the range from three to six, depending on quality requirements [56].

3.3.1 Stopping Conditions

In the preceding discussions of iterative vector approximation and its application to sinusoidal modeling, the recursive nature of analysis-by-synthesis implies that as many (or as few) components as desired may be calculated. The obvious problem that now arises is how to determine the number of components needed to adequately represent a given audio signal.

The simplest approach to the problem would be to set a fixed number of components J, large enough to achieve perceptual identity in all cases of interest. Although this makes sense for certain signals (such as musical tones) whose characteristics are relatively simple, stationary, and well-known in advance, using a set number of components for more complicated audio signals is less effective. For instance, the number of components required to adequately model any given segment of a speech signal can vary widely, depending on voicing state, pitch, and background noise level. As a result, using the minimum number of sinusoids required for a "worst-case" segment to model an entire utterance is very inefficient and results in much unnecessary analysis

computation.

To account for the variable requirements of complicated audio signals, it is useful to derive a local measure of approximation for overlap-add sinusoidal modeling. Note that a by-product of analysis-by-synthesis is the sequential error norm E_ℓ, which may be interpreted as an energy measure of the error between a signal segment and its synthetic approximation as a function of the number of components. Since E_ℓ is known to decrease with increasing ℓ, it makes sense to define a performance measure based on E_ℓ which correlates reasonably well to subjective quality and which can be compared to a threshold value to determine when a given signal segment has been sufficiently approximated.

Perhaps the most widely used measure of modeling performance is the *segmental signal-to-noise ratio* (SNR), defined in terms of frame-based models as the ratio between the energy of a signal segment and its local approximation error energy. Referring to Equation 3.17 and suppressing frame notation, for a given number of components, segmental SNR is defined using E_ℓ as

$$\mathcal{S}[\ell] \triangleq \frac{\|\mathbf{x}\|^2}{E_\ell} = \left(\sum_{n=-N_a}^{N_a} w_a[n] s^2[n + kN_s] \right) / E_\ell, \tag{3.34}$$

and a corresponding decibel measure is given by $\mathcal{S}_{\text{dB}}[\ell] = 10 \log_{10} \mathcal{S}[\ell]$. Since $E_0 = \|\mathbf{x}\|^2$ and since E_ℓ decreases monotonically, $\mathcal{S}_{\text{dB}}[\ell]$ always increases monotonically from a value of $\mathcal{S}_{\text{dB}}[0] = 0$ dB. As a result of this predictable behavior, the segmental SNR $\mathcal{S}_{dB}[\ell]$ is very well-suited to serve as a performance measure in analysis-by-synthesis.

While the exact relationship of segmental SNR to the subjective quality of synthetic audio signals is unclear [57], experimental results indicate that subjective quality improves uniformly as the SNR increases. Therefore, a reasonable approach is to calculate component parameters until $\mathcal{S}[\ell]$ surpasses a given threshold value $\mathcal{S}'_T$ which corresponds to good subjective quality in all cases of interest. For audio signals, informal testing indicates that a value of 30 dB is a good benchmark threshold to achieve perceptual identity for a wide variety of conditions.

Although a high SNR threshold can guarantee high subjective quality while reducing the number of components needed to model many audio signals, a constant-SNR threshold can still result in serious overapproximation. To see this, consider the analysis-by-synthesis example shown in Figure 3.4. In that example the synthetic sequences $\tilde{x}_\ell[n]$ quickly converged to $x[n]$ after only a few iterations; this behavior is typical of quasi-periodic sequences, which are well-modeled by narrowband signals such as sinusoids. In realistic environments, however, analysis-by-synthesis must deal with signals which are not represented so easily. A good example is the background noise often encountered in the "silent" portions of audio signals.

As noted previously, any sequence may be approximated with arbitrary accuracy using analysis-by-synthesis, given enough components. However, the random, wideband character of noise implies that many more sinusoids may be required than for quasi-periodic sequences to achieve similar SNR values. For example, background noise in audio signals sampled at 8 kHz may require as many as 100 components to achieve an SNR value of 30 dB; since low-energy background noise is perceptually unimportant in audio signals, going to such lengths to accurately approximate it is clearly unwarranted.

A qualitative interpretation of the above statement is that while higher SNR levels are needed to accurately model high-energy segments of audio signals, as the signal energy $\|\mathbf{x}\|^2$ decreases the SNR requirement may be relaxed. In other words, rather than using a constant SNR threshold $\mathcal{S}_T'$, in signals with large variations in local energy it makes sense to have an energy-dependent SNR threshold $\mathcal{S}_T(\mathbf{x})$ which decreases as signal energy decreases. The simplest such threshold corresponds to an "energy threshold," where approximation ceases when $E_\ell < E_T$. Substituting Equation 3.34, this condition is equivalent to

$$\mathcal{S}[\ell] > \frac{\|\mathbf{x}\|^2}{E_T} = \mathcal{S}_T(\mathbf{x}). \tag{3.35}$$

Unfortunately, while energy thresholding reduces the effort applied to approximating low-energy signal segments when E_T is set high, the linear dependence of

$\mathcal{S}_T(\mathbf{x})$ on $\|\mathbf{x}\|$ can result in a threshold which is too low for high-energy segments, causing distortion. Conversely, setting E_T low enough to adequately model high-energy segments results in overapproximation of most low-energy segments. For this reason it is desirable to have an energy-dependent threshold with a nonlinear dependence on $\|\mathbf{x}\|$. Since constant-SNR thresholds are more appropriate for high-energy segments and constant-energy thresholds are better for low-energy segments, one approach is to use a piecewise SNR threshold defined by

$$\mathcal{S}_T(\mathbf{x}) = \begin{cases} \mathcal{S}'_T\|\mathbf{x}\|^2/E_T, & \|\mathbf{x}\|^2 \le E_T \\ \mathcal{S}'_T, & \|\mathbf{x}\|^2 > E_T. \end{cases} \tag{3.36}$$

This threshold provides a better balance between the accuracy requirement of high-energy segments and the low numbers of components desired in background noise; however, it also requires deciding whether a given signal segment is background noise based on its energy. To avoid underapproximating important perceptual features because of misclassification, a threshold which does not require this decision would be preferable.

Such a threshold function appears in waveform coding theory. In order to provide consistent signal-to-quantizing noise ratios over a wide dynamic range, "μ-law" companding [58] is often used in scalar quantization. As a function of signal variance σ_x^2, where

$$\sigma_x^2 = E(x^2[n]), \tag{3.37}$$

the average signal-to-quantizing noise ratio in μ-law quantization is given as

$$SNR(\sigma_x) = \mathcal{S}'_T - 10\log_{10}\left[1 + \left(\frac{X_{\max}}{\mu\sigma_x}\right)^2 + \sqrt{2}\left(\frac{X_{\max}}{\mu\sigma_x}\right)\right], \tag{3.38}$$

where $X_{\max}$ is the maximum absolute value of $x[n]$ and μ determines the amount of compression. Figure 3.5 shows plots of $SNR(\sigma_x)$ as a function of $\sigma_x/X_{\max}$ for the case when $\mathcal{S}'_T = 30$ dB and for several values of μ.

As is clear from the plots, $SNR(\sigma_x)$ approaches $\mathcal{S}'_T$ asymptotically as energy increases, and approaches a linear dependence on σ_x as energy decreases, with μ con-

trolling the transition point between the two modes of operation. This is precisely the behavior desired for $\mathcal{S}_T(\mathbf{x})$. The μ-law SNR function of Equation 3.38 is advantageous to use as a threshold, since its parameters correspond to well-understood properties of a given signal such as maximum level and variance, which can easily be adapted to changing environments without requiring trial-and-error adjustment.

As a final point, note that calculating the SNR threshold $\mathcal{S}_T(\mathbf{x})$ for a given signal segment requires the signal variance σ_x rather than $\|\mathbf{x}\|^2$. According to Equation 3.37, this is a statistical quantity which cannot be determined from $x[n]$; however, examining Equation 3.34 reveals that

$$\|\mathbf{x}\|^2 = \sum_{n=-N_a}^{N_a} w_a[n]s^2[n + kN_s]$$

is a biased estimate of σ_x^2. Assuming ergodicity and short-time stationarity, we can then calculate an unbiased estimate of σ_x by

$$\hat{\sigma}_x = \left(\sum_{n=-N_a}^{N_a} w_a[n]s^2[n + kN_s] / \sum_{n=-N_a}^{N_a} w_a[n] \right)^{1/2} \tag{3.39}$$

which is then used in Equation 3.38 to determine an appropriate threshold for a given signal segment. Experimentation on a variety of speech signals indicates that values of μ in the range from 50 to 255 result in accurate synthetic approximation without overapproximating background noise, but that care must be taken not to set μ too low due to "squelching" or suppression of (perhaps desired) low-energy signals.

3.3.2 Frequency-Domain Interpretation

As mentioned before, there is a direct correlation in analysis-by-synthesis between ensemble indices and radian component frequencies. Furthermore, much of the computation required to determine overlap-add sinusoidal model parameters is in the form of inner products between sinusoids of various frequencies and between sinusoids and arbitrary discrete-time sequences. As a result, much of the analysis-by-synthesis algorithm might be expected to be expressible in terms of frequency-domain operations

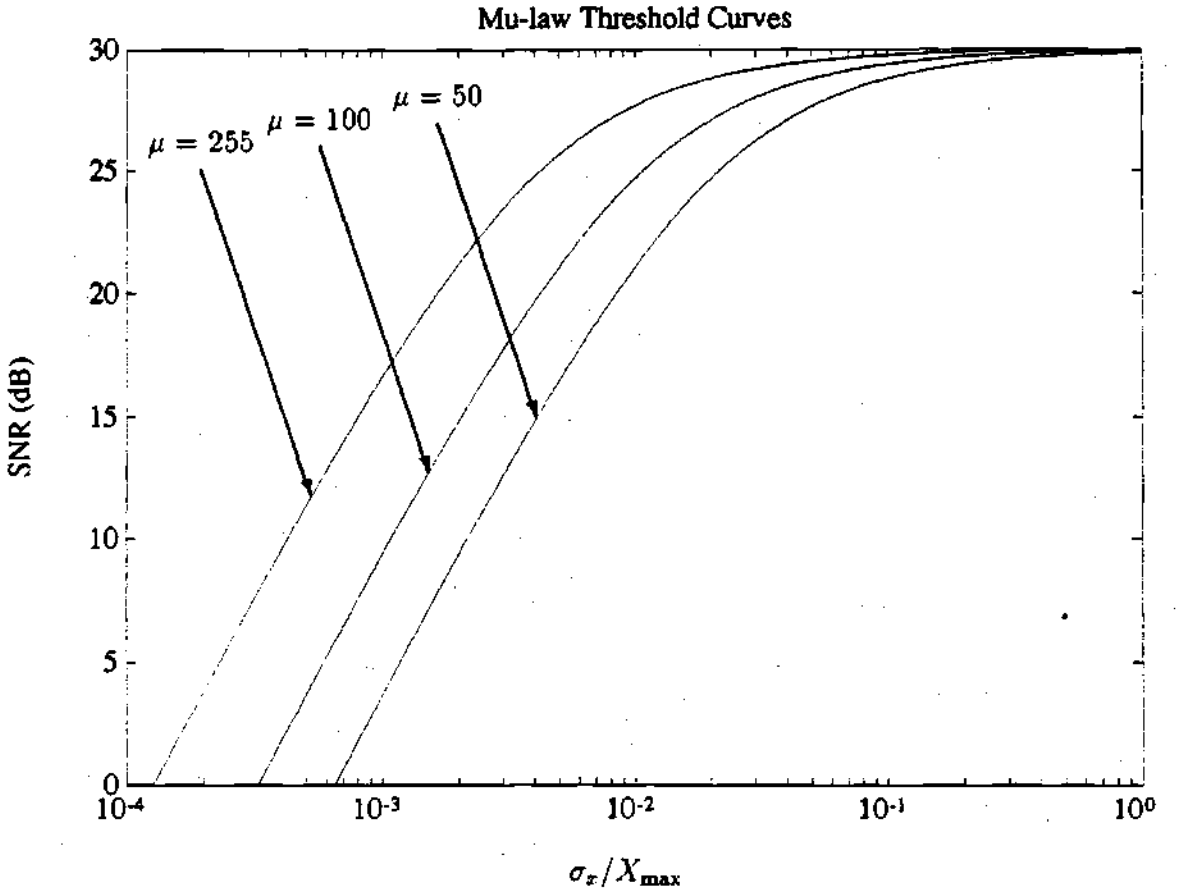

Figure 3.5: Plots of μ-law threshold curves for several values of μ. Note that as μ increases the threshold more closely corresponds to a constant-SNR threshold.

and transforms. This is in fact the case, and as we will see, frequency-domain interpretation of analysis-by-synthesis provides a great deal of useful information for the purposes of design and implementation as well as analysis.

The *discrete-time Fourier transform* (DTFT) of a sequence $x[n]$ is defined by

$$X(e^{j\omega}) \triangleq \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}. \tag{3.40}$$

When $x[n]$ is a real-valued sequence the following identities hold:

$$\begin{aligned} \sum_{n=-\infty}^{\infty} x[n]\cos\omega n &= \Re e\{X(e^{j\omega})\} \\ \sum_{n=-\infty}^{\infty} x[n]\sin\omega n &= -\Im m\{X(e^{j\omega})\}. \end{aligned} \tag{3.41}$$

The DTFT's of the sequences $e_\ell[n]g[n]$, $x[n]g[n]$, $\tilde{x}_\ell[n]g[n]$, $\hat{x}_\ell[n]g[n]$ and $g^2[n]$ will be referred to as $EG_\ell(e^{j\omega})$, $XG(e^{j\omega})$, $\widetilde{X}G_\ell(e^{j\omega})$, $\widehat{X}G_\ell(e^{j\omega})$ and $GG(e^{j\omega})$, respectively.

Equations 3.25 and 3.21 state that the sequential approximation sequence $\tilde{x}_\ell[n]$ and error sequence $e_\ell[n]$ are given recursively by

$$\begin{aligned} \tilde{x}_\ell[n] &= \tilde{x}_{\ell-1}[n] + \hat{x}_\ell[n] \\ e_\ell[n] &= e_{\ell-1}[n] - \hat{x}_\ell[n]; \end{aligned}$$

substituting the first relation into the formula for the DTFT $\widetilde{X}G_\ell(e^{j\omega})$ yields

$$\begin{aligned} \widetilde{X}G_\ell(e^{j\omega}) &= \sum_{n=-N_a}^{N_a} (\tilde{x}_{\ell-1}[n]g[n] + \hat{x}_\ell[n]g[n])e^{-j\omega n} \\ &= \widetilde{X}G_{\ell-1}(e^{j\omega}) + \widehat{X}G_\ell(e^{j\omega}); \end{aligned} \tag{3.42}$$

likewise, $EG_\ell(e^{j\omega})$ may be expressed as

$$EG_\ell(e^{j\omega}) = EG_{\ell-1}(e^{j\omega}) - \widehat{X}G_\ell(e^{j\omega}). \tag{3.43}$$

These relations imply a direct duality between recursively updating the sequential approximation and error sequences and updating spectra associated with these sequences. According to the assumed initial conditions, $EG_0(e^{j\omega}) = XG(e^{j\omega})$ and $\widetilde{X}G_0(e^{j\omega}) = 0$.

The "component spectrum" $\widehat{X}G_\ell(e^{j\omega})$ may be expressed in terms of component parameters A_ℓ, ω_ℓ and ϕ_ℓ using Equation 3.21. Substituting this expression yields

$$\begin{aligned}\widehat{X}G_\ell(e^{j\omega}) &= \sum_{n=-N_a}^{N_a} A_\ell g^2[n]\cos(\omega_\ell n + \phi_\ell)e^{-j\omega n} \\ &= A_\ell \sum_{n=-N_a}^{N_a} g^2[n]\left(\frac{1}{2}e^{j(\omega_\ell n+\phi_\ell)} + \frac{1}{2}e^{-j(\omega_\ell n+\phi_\ell)}\right)e^{-j\omega n}. \end{aligned} \quad (3.44)$$

Letting $\alpha_\ell = \frac{1}{2}A_\ell e^{j\phi_\ell}$, this becomes

$$\begin{aligned}\widehat{X}G_\ell(e^{j\omega}) &= \alpha_\ell \sum_{n=-N_a}^{N_a} g^2[n]e^{-j(\omega-\omega_\ell)n} + \alpha_\ell^* \sum_{n=-N_a}^{N_a} g^2[n]e^{-j(\omega+\omega_\ell)n} \\ &= \alpha_\ell GG(e^{j(\omega-\omega_\ell)}) + \alpha_\ell^* GG(e^{j(\omega+\omega_\ell)}). \end{aligned} \quad (3.45)$$

In other words, the component spectrum $\widehat{X}G(e^{j\omega})$ is simply the conjugate sum of two identical spectra $GG(e^{j\omega})$, the DTFT of $g^2[n]$, shifted left and right by ω_ℓ. This result is due to the *modulation property* of the DTFT [59]. When combined with Equation 3.42 we see that the "approximation spectrum" $\widehat{X}G_\ell(e^{j\omega})$ is simply a weighted sum of shifted versions of $GG(e^{j\omega})$.

Referring to Equation 2.11, the orthogonality conditions of iterative vector approximation for the case of sinusoidal components may be expressed as

$$\sum_{n=-N_a}^{N_a} e_\ell[n]g[n]\cos\omega_\ell n = \sum_{n=-N_a}^{N_a} e_\ell[n]g[n]\sin\omega_\ell n = 0. \quad (3.46)$$

Making use of Equation 3.41, these orthogonality conditions are equivalent to

$$EG_\ell(e^{j\omega_\ell}) = 0; \quad (3.47)$$

that is, under optimal conditions the "error spectrum" obtained by subtracting the component spectrum from the previous error spectrum as in Equation 3.43 will have a *spectral null* at the component frequency ω_ℓ. Another interpretation follows by substituting Equation 3.43:

$$\widehat{X}G_\ell(e^{j\omega_\ell}) = EG_{\ell-1}(e^{j\omega_\ell}). \quad (3.48)$$

This implies that, for a given component frequency ω_ℓ, the amplitude and phase parameters which minimize E_ℓ cause the component spectrum $\widehat{X}G_\ell(e^{j\omega})$ to match the previous error spectrum $EG_{\ell-1}(e^{j\omega})$ at ω_ℓ. Figure 3.6 shows an example of the operation of analysis-by-synthesis from a frequency-domain standpoint, illustrating the interpretations discussed above.

Another interpretation of analysis-by-synthesis in the frequency domain which will prove very useful later relates inner product expressions which must be calculated in analysis-by-synthesis to the spectra defined above. By making use of the identities of Equation 3.41 and the trigonometric relations $\cos^2\theta = \frac{1}{2} + \frac{1}{2}\cos 2\theta$, $\sin^2\theta = \frac{1}{2} - \frac{1}{2}\cos 2\theta$ and $\cos\theta\sin\theta = \frac{1}{2}\sin 2\theta$, the parameters of Equation 3.27 may be expressed as

$$\begin{aligned}
\gamma_{11}^\ell &= \frac{1}{2}\Re e\{GG(e^{j0}) + GG(e^{j2\omega_\ell})\} \\
\gamma_{12}^\ell &= -\frac{1}{2}\Im m\{GG(e^{j2\omega_\ell})\} \\
\gamma_{22}^\ell &= \frac{1}{2}\Re e\{GG(e^{j0}) - GG(e^{j2\omega_\ell})\} \\
\psi_1^\ell &= \Re e\{EG_{\ell-1}(e^{j\omega_\ell})\} \\
\psi_2^\ell &= -\Im m\{EG_{\ell-1}(e^{j\omega_\ell})\}.
\end{aligned} \tag{3.49}$$

Now consider the form of $XG(e^{j\omega})$ and $GG(e^{j\omega})$. According to Equation 3.17, the sequences $x[n]g[n]$ and $g^2[n]$ are given by

$$\begin{aligned}
x[n]g[n] &= w_a[n]\sigma[n+kN_s]s[n+kN_s] \\
g^2[n] &= w_a[n]\sigma^2[n+kN_s].
\end{aligned} \tag{3.50}$$

If it may be assumed that $\sigma[n]$ is constant, as in the case of steady-state voiced speech or other stationary audio signals, then $XG(e^{j\omega})$ is the DTFT of a segment of $s[n]$ multiplied by the analysis window, and $GG(e^{j\omega})$ is simply $W_a(e^{j\omega})$, the DTFT of the analysis window.

Under this "steady state" assumption, the computation involved in analysis-by-synthesis becomes very straightforward. For instance, under the assumption that

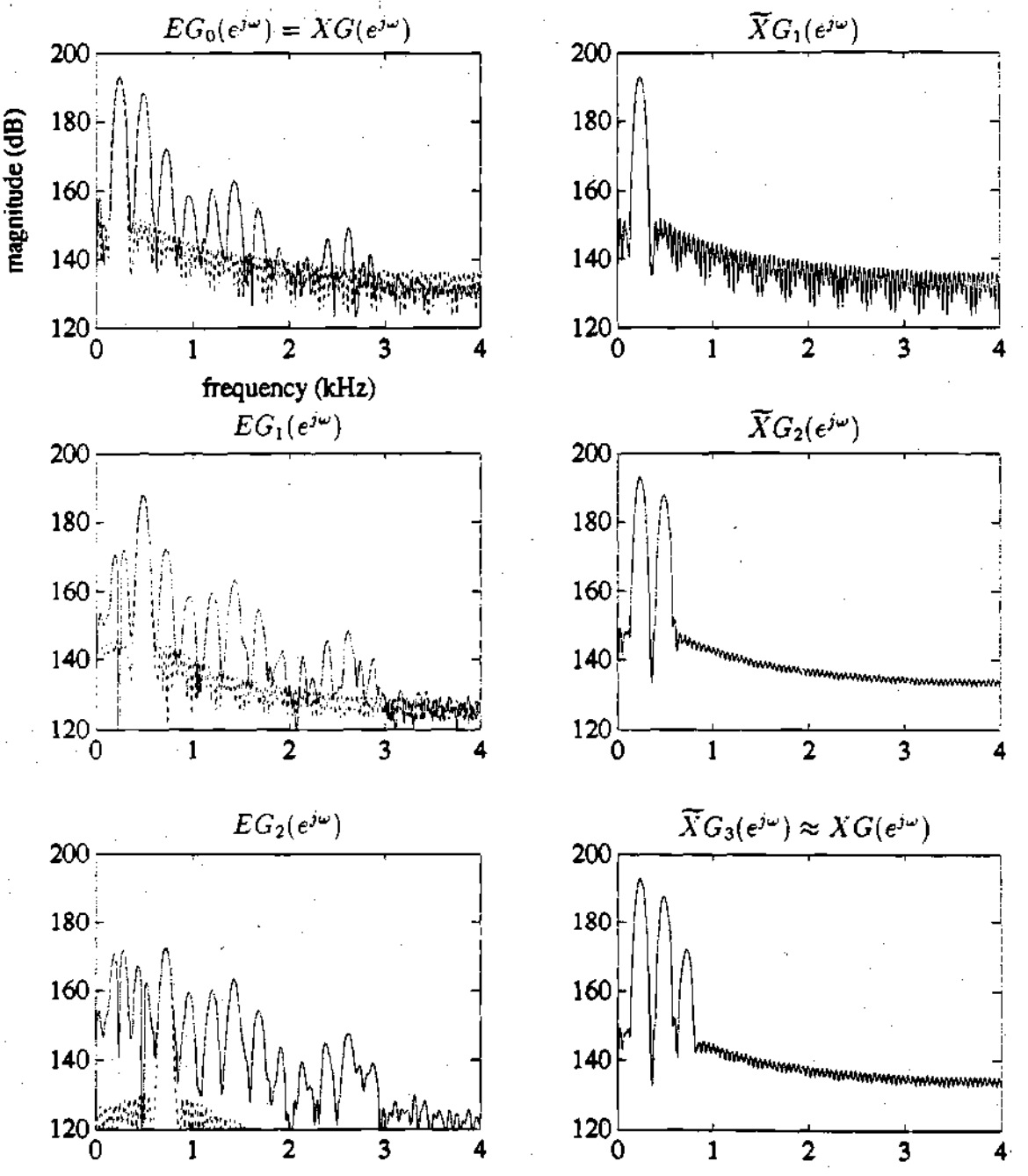

Figure 3.6: Frequency-domain interpretation of analysis-by-synthesis. Left-hand plots show error spectra $EG_\ell(e^{j\omega})$, with optimal component spectra $\widetilde{X}G_\ell(e^{j\omega})$ dotted. Right-hand plots show approximation spectra $\widetilde{X}G_\ell(e^{j\omega})$.

$w_a[n]$ is a symmetric analysis window, $GG(e^{j\omega})$ is real-valued; referring to Equation 3.49, this implies that $\gamma_{12}^{\ell} = 0$. From Equation 3.26, the quadrature parameters a_1^{ℓ} and a_2^{ℓ} are then given by

$$\begin{aligned} a_1^{\ell} &= \psi_1^{\ell}/\gamma_{11}^{\ell} \\ a_2^{\ell} &= \psi_2^{\ell}/\gamma_{22}^{\ell}, \end{aligned} \tag{3.51}$$

and the sequential error norm E_{ℓ}' is expressed as

$$E_{\ell}' = E_{\ell-1}' - \frac{(\psi_1^{\ell})^2}{\gamma_{11}^{\ell}} - \frac{(\psi_2^{\ell})^2}{\gamma_{22}^{\ell}}. \tag{3.52}$$

According to Equation 3.49, γ_{11}^{ℓ} is given in the steady-state case by

$$\gamma_{11}^{\ell} = \{W_a(e^{j0}) + W_a(e^{j2\omega_{\ell}})\}/2.$$

As is well known, tapered windows produce spectra which approximate a frequency-domain impulse centered about $\omega = 0$. As a result, except for small values of ω_{ℓ} it may be assumed that $W_a(e^{j0}) \gg W_a(e^{j\omega_{\ell}})$, hence γ_{11}^{ℓ} and γ_{22}^{ℓ} are approximately equal to $\frac{1}{2}W_a(e^{j0})$. Substituting this approximation into Equation 3.52 yields the result

$$E_{\ell}' \approx E_{\ell-1}' - \frac{|EG_{\ell-1}(e^{j\omega_{\ell}})|^2}{\frac{1}{2}W_a(e^{j0})} \tag{3.53}$$

for most frequency values.

The above results may be used to understand how analysis window choice affects analysis-by-synthesis, in terms of the vector space principles introduced in Chapter 2. Recall from Theorems 2.2 and 2.3 that the performance of iterative vector approximation is best when correlation between component vectors is minimized. To illustrate this point in the context of sinusoidal modeling, consider the problem of approximating the component vector $\hat{\mathbf{x}}_{\ell}$ by projecting onto the space $\boldsymbol{S}_{i_{\ell-1}}$ spanned by the vectors $\mathbf{v}_1^{i_{\ell-1}}$ and $\mathbf{v}_2^{i_{\ell-1}}$. In the steady-state case, recalling Equation 2.18 and substituting the results of Equation 3.51, we have

$$\|\hat{\mathbf{x}}_{\ell}\|^2 = (a_1^{\ell})^2\gamma_{11}^{\ell} + (a_2^{\ell})^2\gamma_{22}^{\ell};$$

substituting the approximations for γ_{11}^{ℓ} and γ_{22}^{ℓ} yields

$$\|\hat{\mathbf{x}}_\ell\|^2 \approx \frac{1}{2}A_\ell^2 W_a(e^{j0}). \tag{3.54}$$

According to Equation 3.53, the error term E_ℓ' in this problem is given by substituting $\|\hat{\mathbf{x}}_\ell\|^2$ for $E_{\ell-1}'$ and $\widehat{X}G_\ell(e^{j\omega_{\ell-1}})$ for $EG_{\ell-1}(e^{j\omega_\ell})$. Combining this with the formula of Equation 2.20 and Definition 2.1 yields

$$\cos[\theta(\hat{\mathbf{x}}_\ell, \mathbf{S}_{i_{\ell-1}})] \approx \frac{|\widehat{X}G_\ell(e^{j\omega_{\ell-1}})|}{\frac{1}{2}A_\ell W_a(e^{j0})}. \tag{3.55}$$

Referring to Equation 3.45, under the steady-state assumption and assuming that $(\omega_\ell + \omega_{\ell-1})$ is reasonably large,

$$\widehat{X}G_\ell(e^{j\omega_{\ell-1}}) \approx \frac{1}{2}A_\ell e^{j\phi_\ell} W_a(e^{j(\omega_{\ell-1}-\omega_\ell)}),$$

and therefore

$$\cos[\theta(\hat{\mathbf{x}}_\ell, \mathbf{S}_{i_{\ell-1}})] \approx \frac{|W_a(e^{j(\omega_{\ell-1}-\omega_\ell)})|}{W_a(e^{j0})}. \tag{3.56}$$

The principal angle cosine given here, which according to Theorem 2.3 should be kept as small as possible to improve the performance of iterative vector approximation, depends only on the difference between $\omega_{\ell-1}$ and ω_ℓ and on the analysis window spectrum $W_a(e^{j\omega})$. As mentioned previously, the spectrum of a tapered window approximates a frequency-domain impulse, with a *mainlobe width* inversely proportional to N_a and *sidelobes* whose magnitudes are independent of N_a. The mainlobe width, as well as the magnitude of accompanying sidelobes, also depends on the window's functional form, with different functions representing different compromises between sidelobe magnitude and mainlobe width [38].

Thus, to keep the magnitude of $\cos[\theta(\hat{\mathbf{x}}_\ell, \mathbf{S}_{i_{\ell-1}})]$ relatively small in cases of interest, it is important to choose an analysis window $w_a[n]$ such that $W_a(e^{j\omega})$ has a narrow mainlobe and sidelobes with relatively small magnitude, and to choose N_a such that the smallest differential frequency expected between any two signal components does not fall in the mainlobe of $W_a(e^{j\omega})$ (known as *mainlobe interference*).

Two good choices in terms of mainlobe width and sidelobe behavior are the *Hamming window*, given by

$$w_a[n] = .54 + .46 \cos \frac{\pi n}{N_a}, \qquad |n| \leq N_a,$$

and the *Kaiser window* (with $\alpha = .5$), described in [60]. Both have mainlobe widths of approximately $4\pi/N_a$ and sidelobe magnitudes less than -40 dB. The Kaiser window has better rolloff properties than the Hamming window, but the Hamming window is considerably simpler to generate and is used henceforth in this work.

Addressing the problem of differential frequencies requires some knowledge of the signal being analyzed. Dealing with signals such as voiced speech and music, which possess a quasi-harmonic structure, it is expected that the minimum frequency between any two signal components is the fundamental frequency ω_o. One reasonable approach to preventing mainlobe interference is to adapt N_a such that the expected average pitch of the signal in question is greater than half the mainlobe width of $W_a(e^{j\omega})$ [4]. For the Hamming window, whose mainlobe width is approximately $4\pi/N_a$ radians, the above requirement is expressed as

$$\overline{\omega}_o > \frac{2\pi}{N_a},$$

or

$$N_a > \frac{2\pi}{\overline{\omega}_o} = \overline{N}_o,$$

where $\overline{N}_o$ is the average pitch period of the signal. In practice, a value of $N_a = 1.25\overline{N}_o$ is used to allow for lower pitch frequencies and for inharmonicity.

Signal Notation Used in This Chapter:

$s_c(t)$: Continuous audio signal

$s[n]$: Sampled audio signal

$\hat{s}[n]$: Synthetic approximation of $s[n]$

$\sigma[n]$: Envelope sequence

$w_s[n]$: Complementary synthesis window

$\hat{s}^k[n]$: Synthetic contribution sequence

$w_a[n]$: Analysis window

$x[n]$: Sequence approximated in analysis-by-synthesis;
$x[n] = (w_a[n])^{1/2} s[n + kN_s]$

$g[n]$: Sequence used to weight component sinusoids in analysis-by-synthesis;
$g[n] = (w_a[n])^{1/2} \sigma[n + kN_s]$

$\hat{x}[n]$: Approximation to $x[n]$ derived from analysis-by-synthesis

$\hat{x}_j[n]$: Component of $\hat{x}[n]$; $\hat{x}_j[n] = A_j g[n] \cos(\omega_j n + \phi_j)$

$\hat{x}_\ell[n]$: Sequential approximation to $x[n]$; $\hat{x}_\ell[n] = \sum_{j=1}^{\ell} \hat{x}_j[n]$

$e_\ell[n]$: Sequential error sequence; $e_\ell[n] = x[n] - \hat{x}_\ell[n]$

$XG(e^{j\omega})$: Spectrum approximated in frequency-domain dual of analysis-by-synthesis;
$XG(e^{j\omega}) = \mathcal{F}\{x[n]g[n]\} = \sum_{n=-\infty}^{\infty} x[n]g[n]e^{-j\omega n}$

$\widehat{X}G_\ell(e^{j\omega})$: "Component spectrum" used to build approximation of $XG(e^{j\omega})$;
$\widehat{X}G_\ell(e^{j\omega}) = \mathcal{F}\{\hat{x}_\ell[n]g[n]\}$

$\widetilde{X}G_\ell(e^{j\omega})$: Sequential approximation spectrum; $\widetilde{X}G_\ell(e^{j\omega}) = \sum_{j=1}^{\ell} \widehat{X}G_j(e^{j\omega})$

$EG_\ell(e^{j\omega})$: Sequential error spectrum; $EG_\ell(e^{j\omega}) = XG(e^{j\omega}) - \widetilde{X}G(e^{j\omega})$

$GG(e^{j\omega})$: "Envelope spectrum" used to construct $\widehat{X}G_\ell(e^{j\omega})$; $GG(e^{j\omega}) = \mathcal{F}\{g^2[n]\}$

$W_a(e^{j\omega})$: Spectrum of analysis window

CHAPTER 4

Application of the ABS/OLA System to Speech Processing

Having introduced the overlap-add sinusoidal model, and having described how analysis-by-synthesis and successive approximation may be combined to provide an effective means of finding appropriate model parameters, what remains is to apply this analysis/synthesis system to problems of interest in signal processing. This chapter discusses some of the relevant issues involved in applying the system in the area of speech processing, and in particular to the problem of speech modification.

4.1 Perceptual Factors in Analysis-by-Synthesis

The analysis-by-synthesis procedure defined in Chapter 3 makes use of a mean-square error norm, which is typical of many approaches to parametric signal modeling. The primary motivations for using this error measure are that it facilitates computation by providing unique, closed-form solutions which can be evaluated numerically, and that it allows analysis in terms of familiar frequency-domain concepts (which, as will be seen later, also enhances computational speed). Also, as previously mentioned, decreasing mean-square error corresponds to increasing subjective quality.

For audio signals, however, mean-square error in the form of segmental SNR does not correlate very well with subjective measures of fidelity perceived by human listeners; as a result, while analysis-by-synthesis as described before can come very close to achieving the highest segmental SNR possible for a given number of compo-

nent sinusoids, this may not imply that perceived quality is as high as possible. The reasons for this behavior are made apparent by considering the analysis-by-synthesis example illustrated in Figures 3.4 and 3.6, and by noting that this example obeys the steady-state assumption made in the previous section. The signal segment being approximated in this example corresponds to voiced speech with a fundamental frequency of approximately 240 Hz; the quasi-harmonic nature of the signal is recognizable in the major spectral peaks of $XG(e^{j\omega})$ shown in Figure 3.6.

Since the perceptually relevant information of quasi-periodic sequences is contained in the form of sinusoidal components with approximately harmonic frequencies, we might expect analysis-by-synthesis to choose components with such frequencies, which is precisely what occurs in the first three iterations of analysis-by-synthesis. Since the segment is assumed steady-state, this behavior is easily understood by examining Equation 3.53: because $E'_{\ell-1}$ and $W_a(e^{j0})$ are fixed values, the optimal component frequency ω_ℓ corresponds to the maximum magnitude of $EG_{\ell-1}(e^{j\omega})$. Referring to Figure 3.6, in the first three iterations this results in choosing the first three harmonic frequencies, as expected.

However, note what occurs when $XG(e^{j\omega})$ is updated after the first iteration to produce $EG_1(e^{j\omega})$: Although the first component spectrum $\widehat{X}G_1(e^{j\omega})$ exactly matches $XG(e^{j\omega})$ at ω_1, there is no guarantee how well it will match the signal spectrum at other frequencies. In particular, note that even though this signal segment is well-modeled as a sum of quasi-harmonic sinusoids, significant spectral energy is left in $EG_1(e^{j\omega})$ in the vicinity of ω_1 corresponding to the mainlobe bandwidth of $W_a(e^{j\omega})$.

This behavior is due to several factors: First, although speech is assumed to be short-time stationary for the sake of defining an overlap-add model, the continuously variable nature of speech production means that no natural speech signal is completely stationary, even in the short term. Also, while estimating the envelope sequence $\sigma[n]$ improves the performance of analysis-by-synthesis in transitory regions, envelope estimation does not perfectly account for all variations of syllabic energy. In addition,

the necessity of searching for an optimal frequency requires a certain amount of error in component frequency estimation. Finally, background noise and the small amount of cross-interference between components can throw analysis off slightly, resulting in residual spectral energy.

In this example, spectral error energy after the third iteration is of higher magnitude near the first three harmonic frequencies than at higher harmonic frequencies. As a result, analysis-by-synthesis will tend to concentrate on reducing this residual spectral error in subsequent iterations rather than choosing components at higher harmonic frequencies. Figure 4.1 shows the amplitudes and frequencies of the first fifteen components determined in analysis-by-synthesis for this example, and an LPC spectral envelope estimate based on the analyzed segment. This figure clearly indicates the tendency of analysis-by-synthesis to "cluster" small amplitude components near high-amplitude components.

Component clustering is undesirable from both a computational and perceptual standpoint. Referring to Figure 4.1, the spectral envelope plot indicates that analysis-by-synthesis has chosen no components in the frequency range of the highest formant near 2.5 kHz; in fact, up to this point analysis-by-synthesis has chosen no components with frequencies higher than 1.5 kHz. Since high-frequency information is perceptually important in audio signals, this behavior implies that more iterations will be necessary to accurately model the speech segment.

To make matters worse, a well-established result from psychoacoustics states that low-amplitude components clustered about high-amplitude components are perceptually irrelevant, since they are "masked" by the larger sinusoids [61]. Clearly, since no perceptual gain is achieved by determining such components, whose analysis adds significantly to the computational load of analysis, steps should be taken in analysis-by-synthesis to discourage clustering.

Because analysis-by-synthesis zealously minimizes mean-square error, regardless of the resulting distribution of component frequencies, one approach to preventing

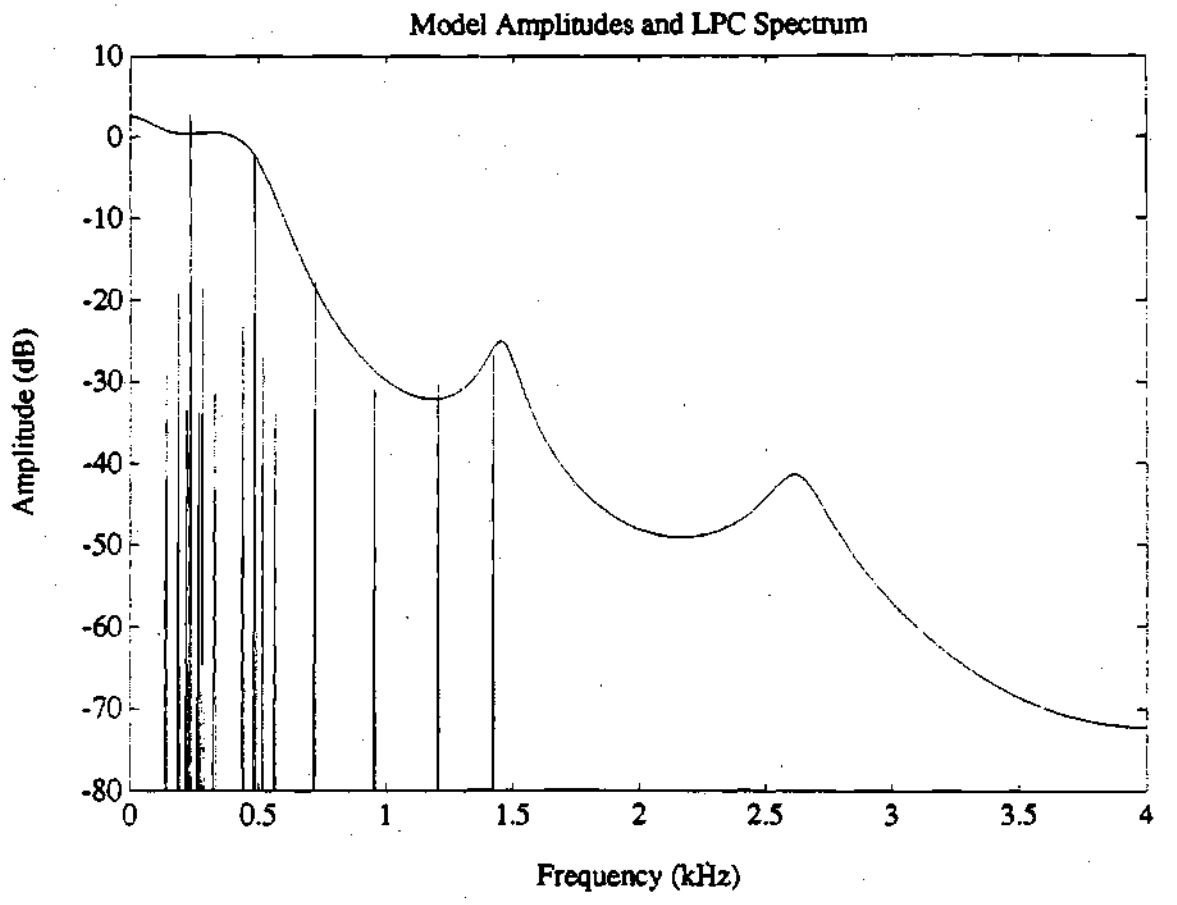

Figure 4.1: Illustration of clustering behavior in analysis-by-synthesis. Note that perceptually insignificant components are determined near major components.

component clustering is to formulate an error measure which is similar to the Euclidean norm but which accounts for the perceptual properties of human hearing. One such error measure, based on the idea of *error spectrum shaping*, has been proposed by Atal and Schroeder [62], and is given in the frequency domain by

$$E_P \triangleq \frac{1}{2\pi}\int_0^{2\pi} |P(e^{j\omega})(S(e^{j\omega}) - \tilde{S}(e^{j\omega}))|^2 d\omega. \tag{4.1}$$

The *perceptual weighting filter* transfer function $P(z)$ is given in terms of the LPC vocal tract transfer function $H(z)$ by

$$P(z) = \frac{H(z/\gamma)}{H(z)}, \tag{4.2}$$

where γ takes values in the range from 0 to 1; a smaller value of γ corresponds to a greater amount of error spectrum shaping.

As shown in Figure 4.2, $P(e^{j\omega})$ has dips in the formant regions of the spectrum and peaks in the inter-formant regions, and this behavior becomes more pronounced for smaller values of γ. Perceptual filtering thus exploits the property of *noise masking* by concentrating error energy near the formant frequencies, where the ear is less sensitive to it. Using Parseval's relation, E_P may be expressed in the time domain as

$$\begin{aligned} E_P &= \sum_{n=-\infty}^{\infty} \{p[n] * (s[n] - \tilde{s}[n])\}^2 \\ &= \sum_{n=-\infty}^{\infty} \{s_P[n] - \tilde{s}_P[n]\}^2. \end{aligned} \tag{4.3}$$

Comparison of E_P with E in Equation 3.15 reveals that the two measures are identical except for the convolution operation in E_P. Furthermore, Equation 4.3 suggests a practical implementation of perceptual weighting, as follows: The input audio signal $s[n]$ is first *prefiltered* by the perceptual weighting filter. Analysis-by-synthesis is then performed using the standard Euclidean norm to produce $\tilde{s}_P[n]$, which is then *postfiltered* by the perceptual weighting filter inverse, yielding $\tilde{s}[n]$. Using this implementation, experiments with speech signals indicate that $\gamma = .4$ yields the best perceptual results, although this value is not critical.

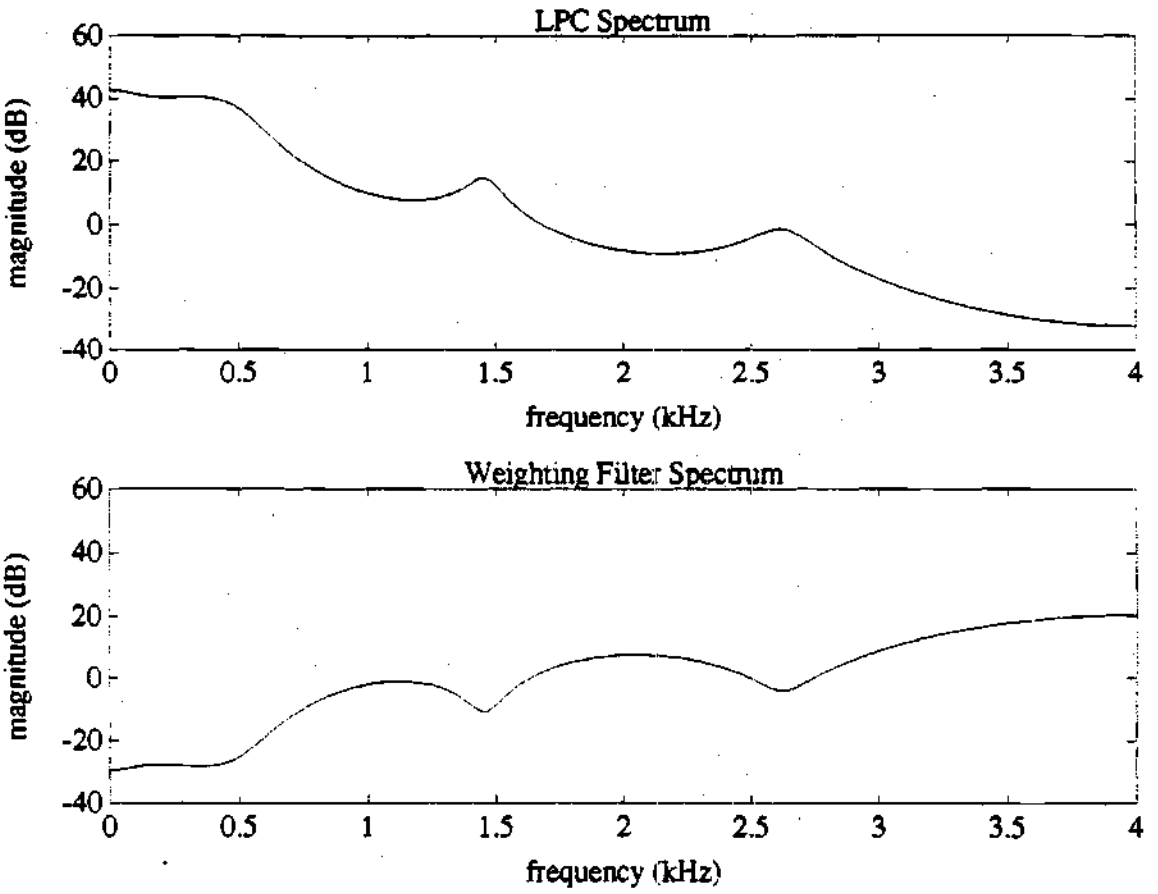

Figure 4.2: An example of the LPC spectral envelope $H(e^{j\omega})$ of a speech segment and its corresponding perceptual weighting filter spectrum $P(e^{j\omega})$ for a value of $\gamma = .4$

The results of perceptual weighting in analysis-by-synthesis have been dramatic, producing both significant improvements in subjective quality over unweighted synthetic speech with a fixed number of components and a considerable reduction in the number of sinusoids required to produce high-quality synthetic speech. Figure 4.3 is a plot of the first fifteen component amplitudes and frequencies used to model a prefiltered version of the speech segment analyzed before. As is evident, perceptual weighting has greatly reduced clustering about major components; while some off-harmonic components are still determined in this case, most components analyzed now correspond to harmonic frequencies. Furthermore, the upper limit of analyzed frequencies has been increased to 2.5 kHz, hence much more high-frequency information has been captured in the approximation.

Unfortunately, perceptual weighting as described above requires applying pre- and post-filters to $s[n]$, which further requires performing LPC analysis to determine $H(z)$; these operations represent a considerable computational overhead in analysis-by-synthesis. Consider now the effect of prefiltering on $s[n]$: According to Equation 4.2, if the original audio signal has $H(e^{j\omega})$ as its spectral envelope, then $s_P[n]$ has a spectral envelope corresponding to $H(e^{j\omega}/\gamma)$. This change of variable implies that poles located at $z = z_o$ in the original LPC transfer function are shifted to $z = \gamma z_o$ in the "prefiltered" transfer function, implying that $H(e^{j\omega}/\gamma)$ becomes "flatter" for smaller values of γ, approaching unity as γ approaches zero. This is seen in the LPC spectral envelope shown in Figure 4.3.

With this in mind it is possible to understand how perceptual weighting works in the context of sinusoidal modeling. As mentioned earlier, the reason clustering occurs is that there is considerable variation in the spectral energy of $s[n]$ in different frequency ranges. Since the prefiltered signal $s_P[n]$ is spectrally flatter than $s[n]$, clustering is reduced; this is because harmonic components at higher frequencies become as important to reducing E_P as those at lower frequencies. Thus, perceptual weighting can be viewed as "equalizing" the audio signal before analysis, eliminating

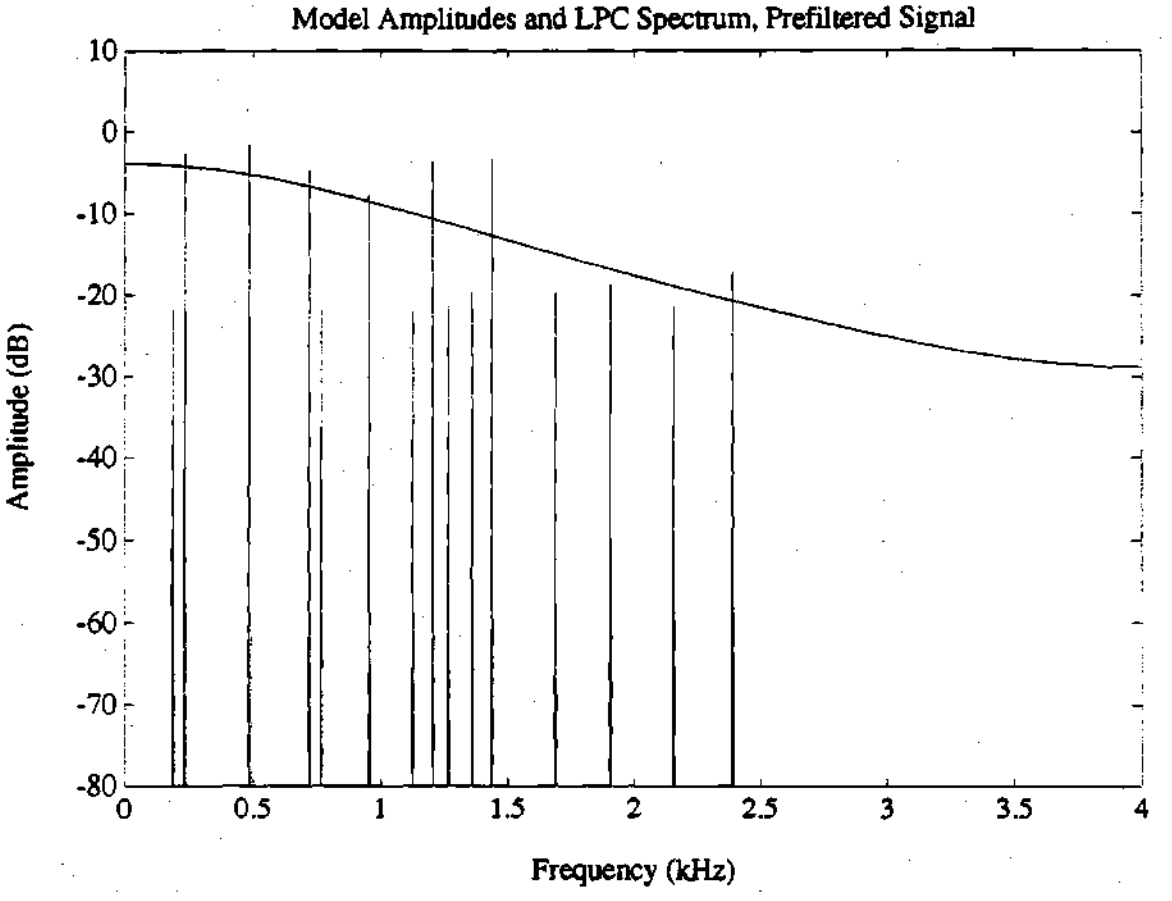

Figure 4.3: Effect of perceptual weighting on analysis-by-synthesis. Note the relative absence of clustering effects compared to Figure 4.1.

frequency-dependent bias.

To achieve this effect, however, it is not always necessary to perform LPC analysis or implement a complicated filter. In the case of speech signals, for instance, there is a natural attenuation in the vocal tract of -6 dB/octave referred to as *spectral tilt* [63] which causes energy to be concentrated in the low-frequency range. Therefore, it is possible to equalize speech signals in the same fashion as perceptual weighting by applying a fixed low-order highpass filter which compensates for spectral tilt, such as that given by

$$s_P[n] = s[n] - .9s[n-1].$$

This simple approach to prefiltering has very little of the computational overhead associated with applying the perceptual weighting filter discussed before, and achieves similar effects. However, the advantage gained is not as pronounced, since removing spectral tilt does not flatten the signal spectrum as much as the perceptual weighting filter. In addition, the simple filter works only with signals (such as speech) which exhibit spectral tilt, and is therefore less general than the more sophisticated approach.

A third, even simpler approach to the problem of clustering is based on the observation made before that masked low-amplitude sinusoids tend to be clustered around a high-amplitude sinusoid mainly in the frequency range corresponding to the mainlobe bandwidth of $W_a(e^{j\omega})$. Since analysis-by-synthesis tends to choose components in order of decreasing energy, if N_a is sufficiently large that perceptually significant components do not interfere then once a component with frequency ω_ℓ is determined, frequencies in the range

$$\omega_\ell - \gamma_b \frac{B_{ml}}{2} \leq \omega \leq \omega_\ell + \gamma_b \frac{B_{ml}}{2},$$

where B_{ml} is the mainlobe bandwidth, are eliminated from the ensemble search thereafter. This "frequency blanking" method, which requires no computational overhead, very effectively reduces clustering, as demonstrated in Figure 4.4. In addition, it

works well for audio signals in general since no *a priori* assumptions of signal properties are required. The parameter γ_b, which controls the amount of frequency blanking, takes values in the range from 0 to 1; $\gamma_b = 0$ corresponds to straightforward analysis-by-synthesis. Experiments with speech signals indicate that $\gamma_b = .75$ yields the best perceptual results, but again this value is not critical.

However, frequency blanking directly affects the operation of analysis-by-synthesis, unlike prefiltering. This is due to pruning the search space, which results in termination of analysis-by-synthesis once all frequencies have been analyzed or eliminated. While this is not a problem if N_a is large enough to meet the non-interference assumption, if N_a is too small then analysis accuracy quickly degrades. Since the computational requirements are significantly less than perceptual weighting and the results superior to simple prefiltering (given a judicious choice of N_a), frequency blanking is the preferred method in this work to account for perceptual factors in speech analysis.

4.2 Quasi-Harmonic Modeling and Fundamental Frequency Tracking

Before proceeding, it is important to introduce a formulation of the overlap-add sinusoidal model which will be useful for modification applications. The sinusoidal model described in Section 3.1 is capable of producing, without constraints, approximations to audio signals that are perceptually identical to the originals; for many applications this is sufficient [64]. However, for the purpose of modifying speech signals it is helpful for the structure of a synthetic contribution to reflect pitch information embedded in the signal. To this end, $\tilde{s}^k[n]$ may be written in *quasi-harmonic* form:

$$\tilde{s}^k[n] = \sum_{j=0}^{J[k]} A_j^k \cos((j\omega_o^k + \Delta_j^k)n + \phi_j^k), \tag{4.4}$$

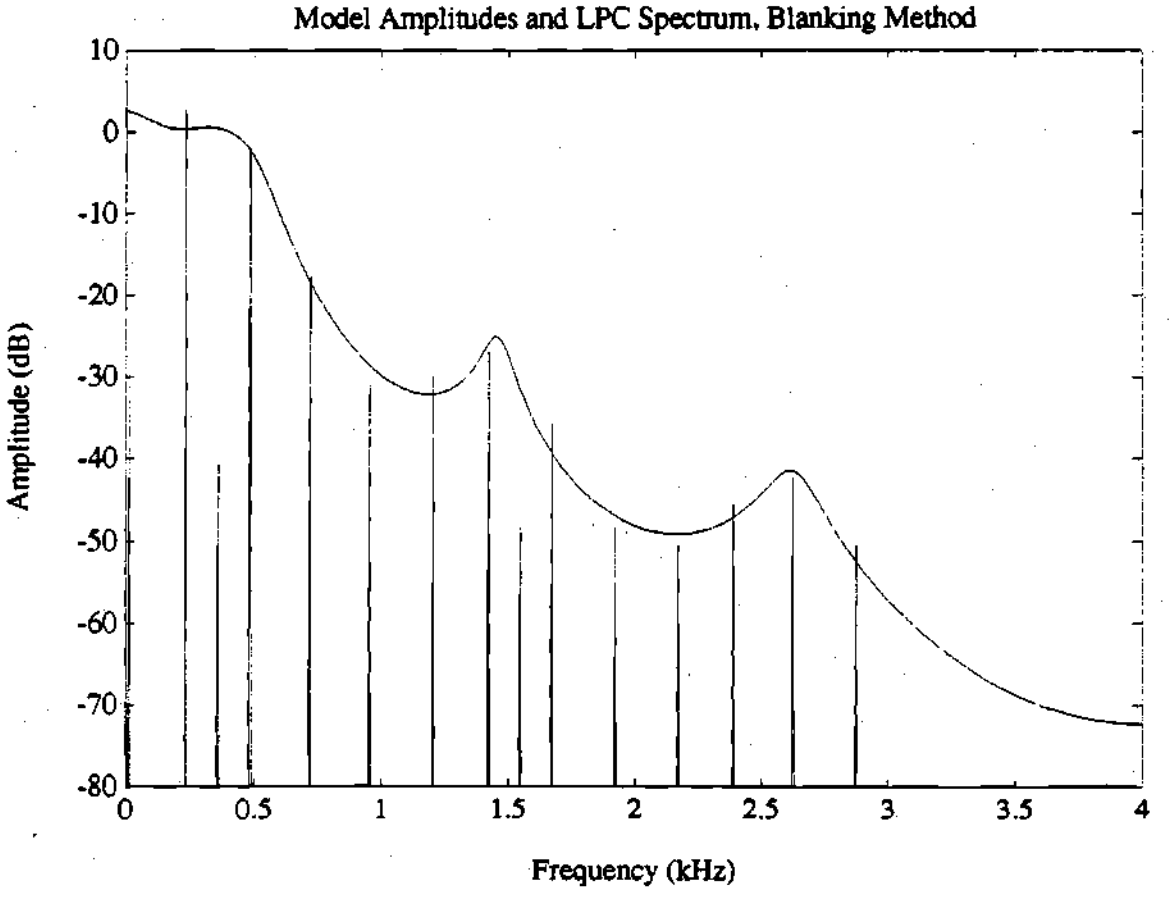

Figure 4.4: Illustration of the effect of frequency blanking to reduce clustering. Note that few spurious components are analyzed.

where $\omega_j^k = j\omega_o^k + \Delta_j^k$, and $J[k]$ is now the greatest integer such that $J[k]\omega_o^k \leq \pi$. Note that only one component is associated with each *harmonic number* j.

With this formulation, it is now necessary to calculate the fundamental frequency $\omega_o^k = 2\pi f_o^k / F_s$ associated with a synthetic contribution as well as the amplitudes, frequencies and phases of $\tilde{s}^k[n]$ in each analysis frame. The problem of estimating the fundamental frequency of speech has been extensively studied; the most popular approaches are based on time-domain processing of speech [65] or LPC excitation signals [66]. However, since the analysis of Section 3.3 provides frequency-domain parameters, and since the estimated fundamental will be employed in a frequency-domain representation, it is reasonable to approach the fundamental frequency estimation problem in the frequency domain as well, using the sinusoidal model parameters determined in analysis.

McAulay and Quatieri have introduced an algorithm which exhaustively evaluates a range of candidate fundamental frequencies in terms of a mean-square error criterion [67]. This approach provides accurate and robust fundamental frequency estimation over a wide range of speech environments, but is also fairly complex. Therefore, a novel algorithm is described here which possesses the same accuracy as, and robustness similar to McAulay and Quatieri's algorithm but which is considerably less complex and hence faster to implement.

Suppose that the component frequencies of $\tilde{s}^k[n]$ in Equation 3.4 are such that the *differential frequencies* $\{\Delta_j^k\}$ are relatively small. In this case, it can be shown [55] that by defining ω_o^k as that value of ω which minimizes the error induced by quantizing frequency parameters to harmonic values,

$$E_F(\omega) = \sum_{n=-N_a}^{N_a} \left\{ \sum_{j=0}^{J[k]} A_j^k [\cos(\omega_j^k n + \phi_j^k) - \cos(j\omega n + \phi_j^k)] \right\}^2 , \qquad (4.5)$$

then ω_o^k is approximately equal to

$$\omega_o^k = \frac{\sum_{i=0}^{J[k]} (iA_i^k)^2 \omega_i^k / i}{\sum_{i=0}^{J[k]} (iA_i^k)^2}, \tag{4.6}$$

assuming that N_a is on the order of a pitch period or larger. Note that this estimate is simply the average of $\{\omega_i^k/i\}$ weighted by $(iA_i^k)^2$.

Again suppressing frame notation, given the sinusoidal model parameters determined using analysis-by-synthesis as in Section 3.3 and an initial fundamental frequency estimate $\omega_o' = 2\pi f_o'/F_s$, it is possible to arrange a subset of the analyzed parameter set in the quasi-harmonic form of Equation 4.4 and to refine the fundamental frequency estimate recursively. This is accomplished by examining the component parameters determined by analysis-by-synthesis in order of increasing j (which, as discussed in Chapter 2, is equivalent to examining the components in order of decreasing energy).

For $1 \leq j \leq J$, the harmonic number associated with ω_j (defined as $\langle \omega_j/\omega_o \rangle$) is calculated. If this number does not conflict with any previous component's harmonic number, the component is included in the quasi-harmonic representation of Equation 4.4, and its amplitude and frequency parameters are used to update ω_o according to Equation 4.6; otherwise, the component is assigned to the set $\mathcal{E}$ of components excluded from the quasi-harmonic representation. For reasons that will become clear, any harmonic numbers left unassigned are associated with zero-amplitude sinusoids at appropriate multiples of the final ω_o. Note that since component energies tend to decrease with increasing j, this process tends to assign the highest energy component possible to each harmonic number, producing the best approximation possible for a given fundamental frequency and enhancing the algorithm's performance in noise [67].

This refinement algorithm presents a paradox when applied to speech signals, since the required initial estimate is usually not available. The above algorithm must therefore be supplemented to determine an appropriate initial estimate. In condi-

tions of low-energy, wideband interference, high-amplitude components correspond to signal components; also, as noted above, higher energy components tend to be determined early in analysis-by-synthesis. Furthermore, much of the energy contained in voiced speech is in the range from 0 to 1000 Hz, and the effects of 60 Hz hum and other low-frequency interference may be avoided without significant loss of information by ignoring frequencies below 100 Hz. Under these conditions, it is reasonable to assume that the highest amplitude component whose frequency is in the range from 100 to 1000 Hz is a signal component, whose frequency $\hat{f}$ is approximately some integer multiple of the actual pitch frequency, *i.e.* $f_o \approx \hat{f}/i$ for some i. Thus, a set of candidate initial fundamental frequency estimates can be derived from $\hat{f}$.

In order to determine an appropriate candidate, a set of values of i are determined such that $f_o'[i] = \hat{f}/i$ falls in the range from 40 to 400 Hz, the typical pitch frequency range for human speech. For each i in this set, the recursive fundamental frequency refinement algorithm is performed using an initial frequency estimate of $\omega_o'[i] = 2\pi f_o'[i]/F_s$. Given the resulting refined estimate $\omega_o[i]$, a measure of the error power induced over the speech analysis frame by fixing the quasi-harmonic frequencies to harmonic values may be derived [55], yielding

$$P_f = \frac{N_a^2}{6}\left(\sum_{j=0}^{J}(A_j\omega_j)^2 - \omega_o[i]\sum_{j=0}^{J}A_j^2 j\omega_j\right). \tag{4.7}$$

Unfortunately, this error measure alone is not sufficient to unambiguously resolve which candidate is best. This is seen by considering the case of a purely periodic signal with fundamental frequency Ω_o; according to Equation 4.7, the frequency-quantization error measure is zero for integer submultiples of Ω_o as well as the fundamental itself. To overcome this inherent ambiguity, a second error measure is needed to resolve which candidate is most appropriate. This second quantity, P_a, results from modeling the quasi-harmonic amplitude parameters using a spectral envelope estimate $H(e^{j\omega})$, and is a measure of the error power induced by excluding components from the quasi-harmonic parameter set and by quantizing the included quasi-harmonic amplitude parameters to a constant multiple of the spectral envelope

magnitude at the component frequencies. The error measure P_a is given by

$$P_a = P_e + \frac{1}{2}\left(\sum_{j=0}^{J} A_j^2 - N_a^2/D_a\right), \tag{4.8}$$

where P_e is the power associated with the components excluded from the quasi-harmonic representation,

$$P_e = \frac{1}{2}\sum_{A_j \in \mathcal{E}} A_j^2, \tag{4.9}$$

and where

$$N_a = \sum_{\ell=0}^{J} A_\ell |H(e^{j\omega_\ell})|, \tag{4.10}$$

$$D_a = \sum_{\ell=0}^{J} |H(e^{j\omega_\ell})|^2. \tag{4.11}$$

At this point a *composite error function* $P_T[i]$ may be constructed as $P_T[i] = P_f + P_a$, and the refined estimate $\omega_o[i]$ corresponding to the minimum value of $P_T[i]$ is chosen as the final estimate ω_o^k.

4.3 Time- and Frequency-Scale Modification

As discussed in Section 1.3, speech modification generally refers to the process of changing some perceptual property of a given speech utterance without affecting other perceptual properties or speech quality. *Time-scale modification*, for instance, refers to changing the articulation rate of speech without changing its fundamental frequency; conversely, *frequency-scale modification* refers to changing fundamental frequency without altering the articulation rate of processed speech, and *pitch-scale modification* adds the constraint that frequency-modified speech must maintain the same short-time spectral envelope as the original. As diescussed in Section 1.3.1, sinusoidal models are well-suited to the task of independently controlling information which corresponds to the perceptual properties of speech. The challenge, then, is to specify techniques for using the proposed ABS/OLA system to perform speech

modifications with the desired perceptual effects and without objectionable artifacts. The algorithms involved in these techniques are by no means simple, and this section discusses them in detail for the cases of time- and frequency-scale modification.

4.3.1 Early Attempts

The Sine-wave Transform System of McAulay and Quatieri performs synthesis using sinusoids whose amplitude and frequency vary in a piecewise continuous manner over time. Time-scale modification can therefore be achieved by varying the rate at which the amplitudes and frequencies change in time, and frequency-scale modification is accomplished by scaling the frequencies of the components without altering their rate of change. In the overlap-add sinusoidal model, however, such continuous parameter functions are not available. With this model, modifications must instead be accomplished by modifying individual synthetic contributions $\tilde{s}^k[n]$ in Equation 3.2 to achieve both the desired changes in time and frequency scale and to maintain phase coherence when the modified contributions are summed together.

As previously mentioned, an approach to performing time- and frequency-scale modification using an overlap-add model formulation (namely the DSTFT) has been reported by Portnoff [32]. By way of analogy to this approach, a simple strategy for performing time- and frequency-scale modification was devised using the quasi-harmonic formulation of the overlap-add sinusoidal model. The technique operates as follows: Referring to Equations 3.6 and 4.4, each synthesis frame is time-scale modified by a factor ρ_k and frequency-scale modified by a factor β_k by changing the synthesis frame length from N_s to $\rho_k N_s$, changing the time scale of the envelope signal $v[n]$ and window signal $w_s[n]$ by ρ_k, scaling the component frequencies of the synthetic signals $\tilde{s}^k[n]$ and $\tilde{s}^{k+1}[n]$ by β_k and β_{k+1}, respectively, and introducing time shifts to the modified synthetic contributions to account for changes in phase coherence due to the modifications. This approach is given quantitatively by the synthesis equation

which generates the modified sequence $\hat{s}[n]$ in the k-th modified synthesis frame:

$$\begin{aligned}\hat{s}[n+N_k] &= \sigma[\frac{n}{\rho_k}+kN_s]\{w[\frac{n}{\rho_k}]\tilde{s}^k[\beta_k(n+\delta^k)] \\ &\quad + w[\frac{n}{\rho_k}-N_s]\tilde{s}^{k+1}[\beta_{k+1}(n+\delta^{k+1}-\rho_kN_s)]\}, \qquad (4.12)\end{aligned}$$

for $0 \leq n < \rho_k N_s$, where $N_k = N_s \sum_{i=0}^{k-1} \rho_i$ is the starting point of the modified synthesis frame. Several points are worth noting here: In order to preserve the complementary nature of overlap-add synthesis, the same time scale factor is applied to both synthesis windows in each synthesis frame. However, in order to prevent frequency discontinuities in the synthetic contributions, frequency-scaling is applied independently to each synthetic contribution in Equation 4.12. Furthermore, the time shift parameter δ^{k+1} may be used as the value of δ^k for the subsequent synthesis frame without phase discontinuity; this allows for a recursive approach to determining time shifts.

The results of this approach were not encouraging, particularly for time or frequency scale factors greater than one. As in the case of modification using the DSTFT, speech modified using this approach tended to have reverberant artifacts as well as a noisy, "rough" quality. Examination of Equations 3.6 and 4.4 reveals the reason for this. By lengthening the synthesis frame to the point where $N_s > N_o$, values of $\tilde{s}^k[n]$ that lie outside the normal analysis frame are used to synthesize the modified speech. Referring to Equation 3.16, the analysis-by-synthesis procedure clearly places no constraints on the behavior of $\tilde{s}^k[n]$ outside the analysis frame. This extrapolation thus yields unpredictable (and hence undesirable) results, as illustrated in Figure 4.5. A more sophisticated model formulation is therefore necessary to achieve higher quality modifications.

4.3.2 A Refined Modification Model

The form of such a model may be derived by making the following observations: In the approach described above it is assumed that the envelope sequence $\sigma[n]$ and the syn-

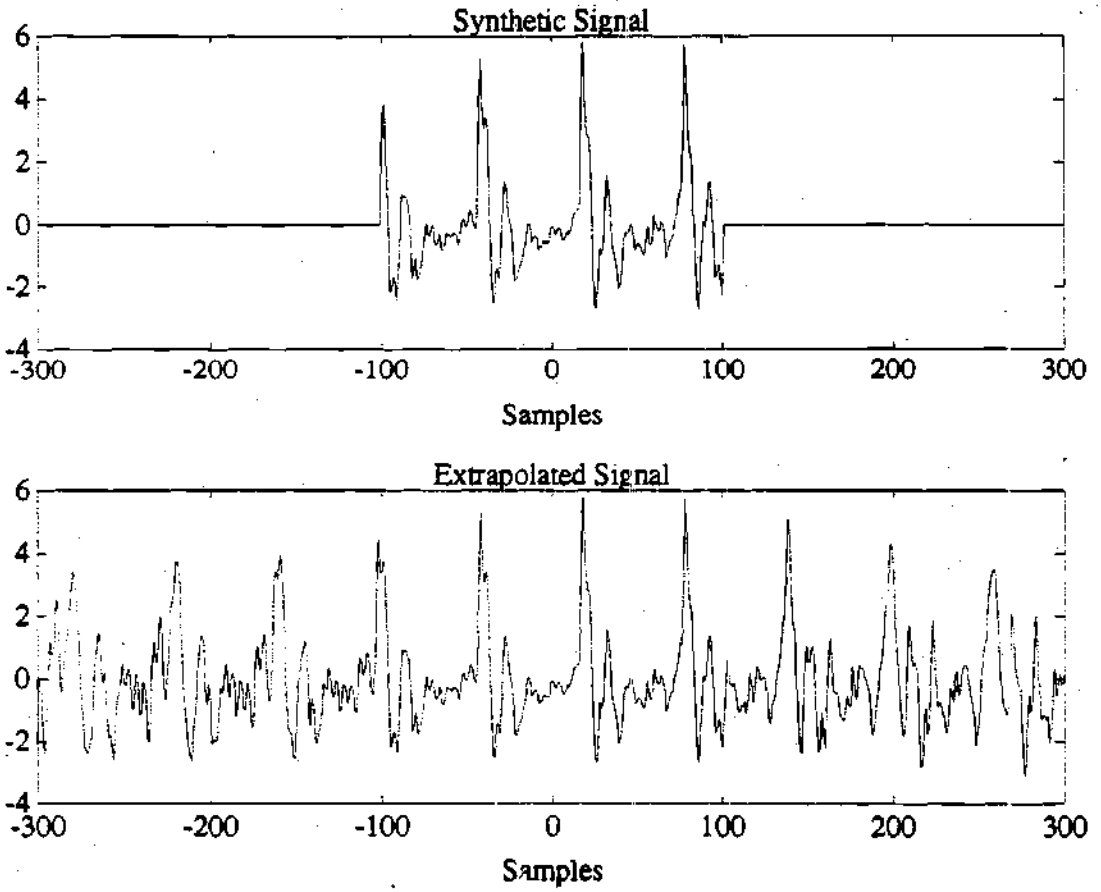

Figure 4.5: Illustration of distortion due to extrapolation beyond analysis frame boundaries ($N_a = 100$). The phase coherence of $\tilde{s}^k[n]$ is seen to break down quickly outside the original analysis frame.

thesis windows account for any slowly-varying amplitude modulation factors involved in the synthesis equation. Slowly-varying amplitude modulation terms correspond to temporal variations in the vocal apparatus during articulation [4]; as a result, the time scale of these modulation terms should be altered during time-scale modification and left unchanged during frequency-scale modification, with no time shift imparted in any case. Furthermore, the synthetic contributions $\tilde{s}^k[n]$ and $\tilde{s}^{k+1}[n]$ are unaltered (save for time shifts) during time-scale modification, and their component frequencies undergo the same multiplicative change for frequency-scale modification.

But consider the *rotating phasor form* of $\tilde{s}^k[n]$ as given in Equation 4.4, which may be written as

$$\begin{aligned} \tilde{s}^k[n] &= \Re e\{\sum_{\ell=0}^{J[k]} A_\ell^k e^{j((\ell\omega_o^k+\Delta_\ell^k)n+\phi_\ell^k)}\} \\ &= \Re e\{\sum_{\ell=0}^{J[k]} \alpha_\ell^k e^{j\Delta_\ell^k n} e^{j\ell\omega_o^k n}\}, \end{aligned} \tag{4.13}$$

where $\alpha_\ell^k = A_\ell^k e^{j\phi_\ell^k}$. Since the differential frequencies $\{\Delta_j^k\}$ are assumed to be relatively small, each complex exponential factor $e^{j\Delta_\ell^k n}$ may be viewed as a slowly-varying amplitude modulation term acting independently on each harmonic component $\alpha_\ell^k e^{j\ell\omega_o^k n}$. Based on the previous discussion of the role of amplitude modulation in modification, the time scale of these terms should therefore be altered during time-scale modification and left unchanged during frequency-scale modification, with no time-shift imparted in either case. The harmonic frequencies are then scaled by the factor β_k to produce a fundamental frequency change, and a global time shift parameter is imparted to harmonic components in order to preserve temporal phase coherence in the modified speech.

These observations may be used to construct a synthesis equation in the context of time- and frequency-scale modification which is similar to Equation 4.12:

$$\hat{s}[n+N_k] = \sigma[\frac{n}{\rho_k} + kN_s]\{w[\frac{n}{\rho_k}]\tilde{s}^k_{\rho_k,\beta_k}[n] + w[\frac{n}{\rho_k} - N_s]\tilde{s}^{k+1}_{\rho_k,\beta_{k+1}}[n-\rho_k N_s]\}. \tag{4.14}$$

where

$$\tilde{s}^k_{\rho_k,\beta_k}[n] = \sum_{j=0}^{J[k]} A_j^k \cos(j\beta_k\omega_o^k(n+\delta^k) + \frac{\Delta_j^k n}{\rho_k} + \phi_j^k)$$

$$\tilde{s}^{k+1}_{\rho_k,\beta_{k+1}}[n] = \sum_{j=0}^{J[k+1]} A_j^{k+1} \cos(j\beta_{k+1}\omega_o^{k+1}(n+\delta^{k+1}) + \frac{\Delta_j^{k+1} n}{\rho_k} + \phi_j^{k+1}). \quad (4.15)$$

This approach to modification is very similar in form to the strategy used in DSTFT-based modification, but with a very important difference: the component frequencies of synthetic contributions are altered according to the relation

$$\hat{\omega} = j\beta\omega_o + \Delta_j/\rho. \quad (4.16)$$

This implies that as the time scale factor ρ is increased, component frequencies tend to "pull in" towards the harmonic frequencies, and in the limit the synthetic contributions become purely periodic sequences.

To understand this behavior, consider the effect of differential frequency terms on the *intra-frame coherence* of harmonic components. Since the differential frequencies are independent of one another, they cause the phase of each component sinusoid to evolve nonuniformly with respect to other components, resulting in a breakdown of coherence in $\tilde{s}^k[n]$ as the time index deviates beyond analysis frame boundaries, as shown in Figure 4.5. By scaling the differential frequencies according to Equation 4.16, this *phase evolution* is slowed, so that intra-frame coherence breaks down proportionally farther from the frame center to account for the longer synthesis frame length. This behavior is illustrated in Figure 4.6 for the case when $\rho = 3$, $\beta = 1$.

At this point what remains is to specify the time shifts δ^k and δ^{k+1} of the modified synthetic contributions. These time shifts may be specified in terms of constraints designed to preserve the coherence of synthetic contributions from frame to frame. To quantify *inter-frame coherence*, we begin with the pitch onset time model of the glottal excitation waveform $e[n]$ described in Section 1.3.1, which is based on a spectral envelope estimate $H^k(e^{j\omega})$ and the quasi-harmonic sinusoidal representation of Equations 3.2 and 4.4 [39]. As described in Section 1.3.1, the synthetic contributions

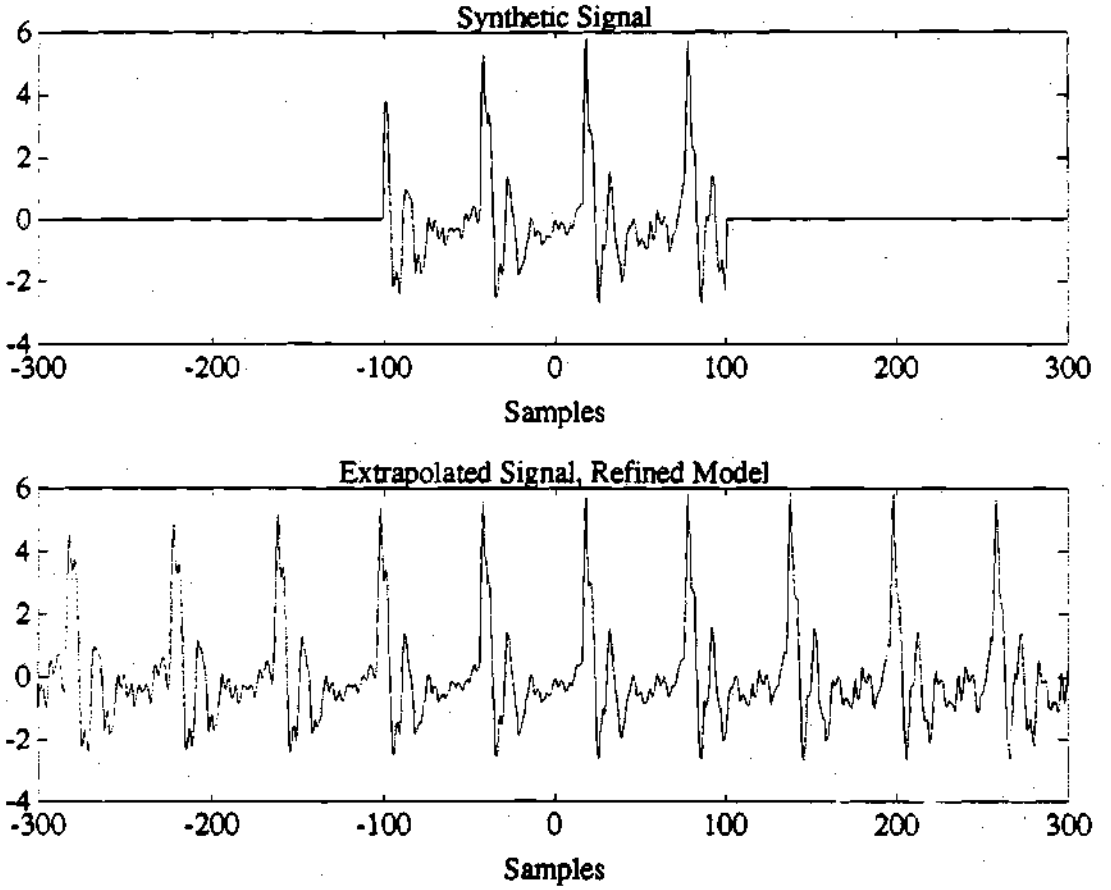

Figure 4.6: Illustration of the effect of differential frequency scaling in the refined modification model. Phase coherence breaks down more slowly in this model due to "pulling in" the differential frequencies (cf. Fig. 4.5).

$\tilde{s}^k[n]$ are replaced by contributions of the form

$$\tilde{e}^k[n] = \sum_{\ell=0}^{J[k]} b_\ell^k \cos(\omega_\ell^k n + \theta_\ell^k), \tag{4.17}$$

where the amplitude and phase parameters of $\tilde{e}^k[n]$ are given by

$$\begin{aligned} b_\ell^k &= A_\ell^k / |H(e^{j\omega_\ell^k})| \\ \theta_\ell^k &= \phi_\ell^k - \angle H(e^{j\omega_\ell^k}). \end{aligned} \tag{4.18}$$

These operations act to remove the effects of the vocal tract in the sense of frequency-domain deconvolution [42]. Assuming for simplicity that $\omega_\ell^k = \ell\omega_o^k$ and suppressing frame notation, Equation 4.17 may be rewritten as

$$\tilde{e}[n] = \sum_{\ell=0}^{J} b_\ell \cos(\ell\omega_o(n - \tau_p) + \psi_\ell(\tau_p)), \tag{4.19}$$

where

$$\psi_\ell(\tau_p) = \theta_\ell + \ell\omega_o\tau_p. \tag{4.20}$$

Since the glottal excitation waveform of voiced speech is expected to correspond approximately to a pulse train whose fundamental frequency varies with changes in pitch, each synthetic contribution to the excitation waveform may be expected to correspond approximately to a periodic pulse train with fundamental frequency ω_o^k. As a result, the representation of Equation 4.19 will have the property that for some value of τ_p (referred to as the *pitch onset time*), the "time-shifted" phase parameters $\{\psi_\ell(\tau_p)\}$ will all be close to either zero or π, or *maximally coherent*. Based on this phenomenon, a technique for estimating the pitch onset time parameter is proposed here which is an extension of the technique reported by McAulay and Quatieri in [39].

For any given value of τ_p in Equation 4.19, an approximation to $\tilde{s}^k[n]$ may be constructed by assuming the condition of maximal coherence and reversing the deconvolution process of Equation 4.18, yielding

$$\tilde{s}_{\tau_p}^k[n] = \sum_{\ell=0}^{J[k]} A_\ell^k \cos(\ell\omega_o^k(n - \tau_p) + \angle H(e^{j\omega_\ell^k}) + m\pi), \tag{4.21}$$

where m is either zero or one. Note that $\tilde{s}^k_{\tau_p}[n]$ may be viewed in this context as the signal produced by driving the vocal tract filter with a pulse train offset from the time origin by τ_p samples. The pitch onset time parameter τ_p may then be formally defined as that value of τ which yields the minimum mean-square error between $\tilde{s}^k[n]$ and $\hat{s}^k_\tau[n]$,

$$E_T(\tau) = \sum_{n=-N_a}^{N_a} \left\{ \tilde{s}^k[n] - \sum_{\ell=0}^{J[k]} A^k_\ell \cos(\ell\omega^k_o(n-\tau) + \angle H(e^{j\omega^k_\ell}) + m\pi) \right\}^2 . \qquad (4.22)$$

Since N_a is typically greater than a pitch period, this is approximately equivalent to finding the absolute maximum of the pitch onset likelihood function

$$L(\tau) = \sum_{\ell=0}^{J} A^2_\ell \cos(\psi_\ell(\tau)) \qquad (4.23)$$

in terms of τ. Unfortunately, this problem does not have a closed-form solution; however, due to the form of $\psi_\ell(\tau)$, $L(\tau)$ is periodic with period $2\pi/\omega_o$. Therefore, the pitch onset time may be estimated by evaluating $L(\tau)$ at uniformly spaced points on the interval $[-\pi/\omega_o, \pi/\omega_o]$ and choosing τ_p to correspond to the maximum of $|L(\tau)|$.

As previously mentioned, the ideal glottal excitation waveform for voiced speech is a variable-frequency pulse train; to produce such a structured waveform using an overlap-add model, the pulse locations of the synthetic contributions given by Equation 3.4 must be highly correlated from one frame to the next. Therefore, in the presence of modifications this correlation must be maintained if the resulting modified voiced speech is to be free from artifacts. To accomplish this, the time shifts δ^k and δ^{k+1} in Equation 4.15 may be determined such that the underlying excitation signal obeys specific constraints in both the unmodified and modified cases [40]. In the ABS/OLA system this is done using an extension of the shape-invariant modification coherence algorithm of the STS, which is described next.

As the name implies, the pitch onset time τ_p represents the location of a pitch pulse in the excitation waveform relative to the synthesis frame boundary. As illustrated in Figure 4.7, by examining Equations 4.19 and 3.6 it is clear that in synthesis

frame k the k-th unmodified synthetic contribution to the excitation, $\tilde{e}^k[n]$, has pulse locations relative to the left frame boundary ($n - kN_s = 0$) given by

$$t_{\mathrm{pl}}^k[i] = \tau_p^k + iT_o^k, \tag{4.24}$$

where $T_o^k = 2\pi/\omega_o^k$. These pulse locations are denoted by O's. Likewise, the pulse locations of $\tilde{e}^{k+1}[n]$ relative to the right frame boundary are given by

$$t_{\mathrm{pl}}^{k+1}[i] = \tau_p^{k+1} + iT_o^{k+1}; \tag{4.25}$$

these pulses are denoted by X's. As shown in Figure 4.7, for some integer i_k a pulse location of $\tilde{e}^k[n]$ is adjacent to the center of the frame (denoted by the dotted line); similarly, for some i_{k+1} a pulse location of $\tilde{e}^{k+1}[n]$ is adjacent to the frame center. The values of i_k and i_{k+1} can be found as[1]

$$\begin{aligned} i_k &= \left\lfloor (N_s/2 - \tau_p^k)/T_o^k \right\rfloor \\ i_{k+1} &= \left\lfloor -(N_s/2 + \tau_p^{k+1})/T_o^{k+1} \right\rfloor + 1. \end{aligned} \tag{4.26}$$

The time difference between these adjacent pulses is given by

$$\Delta = t_{\mathrm{pl}}^k[i_k] - t_{\mathrm{pl}}^{k+1}[i_{k+1}] + N_s, \tag{4.27}$$

and is typically somewhere between T_o^k and T_o^{k+1} for voiced speech.

In the presence of time- and frequency-scale modification, modified speech is produced by overlap-add synthesis using the modified contributions given in Equation 4.15. By frequency-domain deconvolution as in Equation 4.18, this results in modified synthetic excitation contributions[2] of

$$\begin{aligned} \tilde{e}_{\rho_k,\beta_k}^k[n] &= \sum_{j=0}^{J[k]} b_j^k \cos(j\beta_k\omega_o^k(n+\delta^k) + \theta_j^k) \\ \tilde{e}_{\rho_k,\beta_{k+1}}^{k+1}[n] &= \sum_{j=0}^{J[k+1]} b_j^{k+1} \cos(j\beta_{k+1}\omega_o^{k+1}(n+\delta^{k+1}) + \theta_j^{k+1}). \end{aligned} \tag{4.28}$$

[1] $\lfloor \cdot \rfloor$ denotes the "greatest integer less than or equal to" operator.

[2] Assuming zero differential frequencies.

Recalling from Equation 4.20 that $\theta_j^k = \psi_j(\tau_p) - j\omega_o^k \tau_p^k$, this implies that the modified excitation contributions have pulses located at

$$\begin{aligned} \hat{t}_{\mathrm{pl}}^k[i] &= \frac{\tau_p^k}{\beta_k} - \delta^k + i\frac{T_o^k}{\beta_k} \\ \hat{t}_{\mathrm{pl}}^{k+1}[i] &= \frac{\tau_p^{k+1}}{\beta_{k+1}} - \delta^{k+1} + i\frac{T_o^{k+1}}{\beta_{k+1}} \end{aligned} \tag{4.29}$$

relative to the left and right frame boundaries, respectively. As in the unmodified case, pulses from the modified excitation contributions are adjacent to the modified frame center for $i = \hat{\imath}_k$ and $i = \hat{\imath}_{k+1}$, respectively. Assuming that the value of δ^k is known beforehand, δ^{k+1} is the only free parameter left in the synthesis equations and can be used to adjust the time difference between the adjacent pulses.

$$\hat{\Delta} = \hat{t}_{\mathrm{pl}}^k[\hat{\imath}_k] - \hat{t}_{\mathrm{pl}}^{k+1}[\hat{\imath}_{k+1}] + \rho_k N_s. \tag{4.30}$$

In order to ensure that the overall modified excitation waveform exhibits no unexpected fluctuations in pitch period in the presence of modifications, the time shift δ^{k+1} must be adjusted such that $\hat{\Delta}$ is altered by frequency scale modification in the same way as the excitation contributions. Since the frequency scale factors β_k and β_{k+1} may be different, a reasonable constraint is to require that $\hat{\Delta} = \Delta/\beta_{\mathrm{av}}$ as shown in Figure 4.7, where $\beta_{\mathrm{av}} = (\beta_k + \beta_{k+1})/2$. Substituting the values of Δ and $\hat{\Delta}$ from Equations 4.27 and 4.30, this requirement leads to an equation which can be solved recursively for δ^{k+1},

$$\begin{aligned} \delta^{k+1} &= \delta^k + (\rho_k - 1/\beta_{\mathrm{av}})N_s + \frac{\beta_k - \beta_{k+1}}{2\beta_{\mathrm{av}}}\left(\frac{\tau_p^k}{\beta_k} + \frac{\tau_p^{k+1}}{\beta_{k+1}}\right) - \frac{\hat{\imath}_k}{\beta_k}T_o^k \\ &\quad + (i_k T_o^k - i_{k+1}T_o^{k+1})/\beta_{\mathrm{av}}, \end{aligned} \tag{4.31}$$

where

$$\hat{\imath}_k = \left\lfloor (\beta_k(\delta^k + \rho_k N_s/2) - \tau_p^k)/T_o^k \right\rfloor. \tag{4.32}$$

An important point to make concerning this algorithm relates to its performance in the presence of common fundamental frequency estimation errors. Consider the

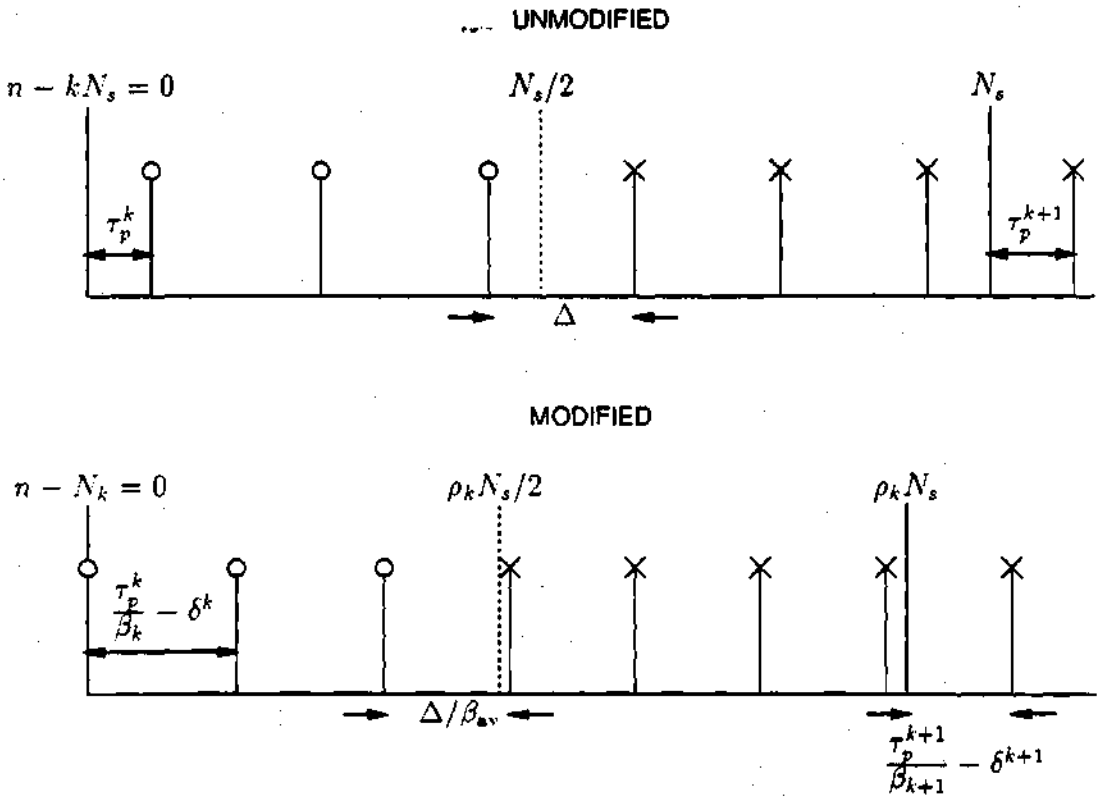

Figure 4.7: Illustration of inter-frame coherence preservation algorithm.

case when the estimated fundamental frequency in frame k is the actual fundamental divided by an integer factor m. The true adjacent pulse spacing Δ of Equation 4.27 can be expressed as $\Delta = T_o^k + \epsilon$. Since the only effect of the fundamental frequency error on pitch onset time estimation is to increase the number of candidate onset times, it may be assumed that no error is induced in the resulting value of τ_p^k. Due to the erroneous pitch estimate, successive pulses in frame k are now spaced by mT_o^k; since τ_p^k is accurate, the erroneous adjacent pulse spacing assumed in the coherence algorithm is expressible as $\Delta_\epsilon = lT_o + \epsilon$ for some l. In the presence of time- and frequency-scale modification, this spacing becomes $\hat{\Delta}_\epsilon = lT_o/\beta + \epsilon/\beta$.

Since the excitation waveform is periodic with period T_o^k/β when modified, the integer l becomes irrelevant; it may thus be set to one, yielding

$$\hat{\Delta}_\epsilon = T_o/\beta + \epsilon/\beta = \hat{\Delta};$$

a similar argument may be invoked to show that the case of a fundamental frequency estimate which is an integer multiple of the actual frequency produces the same result, provided pitch onset time estimation is robust. These examples demonstrate the ability of this coherence preservation algorithm to correct itself in the presence of the most common types of fundamental frequency estimation errors encountered. Since pitch estimators will always make such gross errors, this result is extremely important in terms of modified speech quality.

4.4 Pitch-Scale Modification

While frequency-scale modification successfully changes the fundamental frequency of analyzed speech without changing its time scale or introducing artifacts, when applied to pitch alteration frequency-scale modification has significant disadvantages. Referring to Equation 4.4, the radian frequencies of components in the quasi-harmonic model are seen to span the frequency range $[0, \pi]$; when frequency scale factors greater than one are used in modification, the modified component frequencies of higher

harmonics can fall outside this range, resulting in aliasing. For this reason it is necessary to set the amplitude of any component whose modified frequency is greater than π to zero, resulting in a loss of information.

Conversely, when scale factors less than one are used, the modified component frequencies span a proportionally smaller range, resulting in a loss of high-frequency energy and imparting a "muffled" quality to the modified speech. In addition to these difficulties, frequency-scale modification can seriously degrade the intelligibility of modified speech. In frequency-scale modification, the component amplitudes $\{A_j^k\}$ are unaltered; as a result, changing the component frequencies compresses or expands the short-time spectral envelope of speech for scale factors less or greater than one, respectively. Since both message content and speaker identity depend critically on the spectral envelope, this spectral distortion is undesirable. For these reasons it is important to consider an approach to pitch-scale modification, which attempts to change the fundamental frequency of analyzed speech while preserving its spectral envelope.

One approach to pitch-scale modification was proposed in the context of the Sine-wave Transform System in [5] and [42]. As discussed in the previous section, the glottal excitation waveform may be represented using a sinusoidal model whose parameters are determined by the relations of Equation 4.18. Since the excitation waveform is spectrally flat, the approach suggested by Quatieri and McAulay is to first frequency-scale modify the excitation signal, then to "reconvolve" the excitation parameters with the spectral envelope estimate $H(e^{j\omega})$ at the modified frequencies.

While this approach succeeds in preserving the spectral envelope of modified speech, several problems are apparent. Since the excitation signal is frequency-scale modified, the problems of aliasing and high-frequency energy loss are not addressed, implying that modified speech still sounds muffled. In addition, deconvolving the signal to produce the excitation implies amplification of noise in low-energy portions of the spectrum; when the excitation signal is modified and reconvolved, any noise

added to components which began in an inter-formant region and ended up near a formant will therefore be highly amplified; this noise amplification can seriously affect the quality of pitch-modified speech. What follows is a description of an alternate approach to pitch-scale modification which addresses these problems using the quasi-harmonic sinusoidal model.

As noted above, frequency-scale modification of the excitation sequence results in a loss of high-frequency energy in pitch-scale modification for scale factors less than one. To address this problem, consider a single synthetic contribution to $e[n]$:

$$\tilde{e}^k[n] = \sum_{\ell=0}^{J[k]} b_\ell^k \cos(\ell\omega_o^k n + \Delta_\ell^k n + \theta_\ell^k). \tag{4.33}$$

The objective in modifying the fundamental frequency of $\tilde{e}^k[n]$ without information loss is to specify a set of amplitude, differential frequency and phase parameters for a modified excitation contribution similar in form to $\tilde{e}_{1,\beta_k}^k[n]$ in Equation 4.28, given by

$$\hat{e}_{\beta_k}^k[n] = \sum_{\ell=0}^{J[k]} \hat{b}_\ell^k \cos(\beta_k \ell\omega_o^k(n+\delta^k) + \hat{\Delta}_\ell^k n + \hat{\theta}_\ell^k), \tag{4.34}$$

such that the component frequencies of $\hat{e}_{\beta_k}^k[n]$ span the range $[0,\pi]$. Since as a function of frequency the pairs of amplitude and phase parameters are evenly spaced, a reasonable approach to this problem is to interpolate the complex *phasor form* of the unmodified amplitude and phase parameters in the frequency domain and to resample this interpolated function at modified frequencies to derive the parameters of Equation 4.34. In other words, given the interpolated function $\mathcal{E}(\omega)$, where

$$\mathcal{E}(\omega) = \sum_{\ell=0}^{J} b_\ell e^{j\theta_\ell} I(\omega - \ell\omega_o), \tag{4.35}$$

and where frame notation is again suppressed, the modified amplitudes are given by $\hat{b}_\ell = |\mathcal{E}(\beta\ell\omega_o)|$, and the modified phases by $\hat{\theta}_\ell = \angle\mathcal{E}(\beta\ell\omega_o)$.

While any interpolation function $I(\omega)$ with the properties $I(\ell\omega_o) = 0$ for $\ell \neq 0$ and $I(0) = 1$ may be employed, interpolation using a bandlimited function such as a

raised-cosine of the form

$$I(\omega) = \begin{cases} \cos^2(\pi\omega/2\omega_o), & |\omega| \leq \omega_o \\ 0, & \text{otherwise.} \end{cases} \tag{4.36}$$

is usefule. Such an interpolator makes the computation of $\mathcal{E}(\omega)$ simpler, since all but two terms drop out of Equation 4.35 at any given frequency. Furthermore, since $I(\omega)$ is bandlimited, the effect of any single noise-corrupted component of $\tilde{e}^k[n]$ on the modified parameters is strictly limited to the immediate neighborhood of that component's frequency. This greatly reduces the problem of noise amplification by assuring that noise effects in one part of the speech spectrum do not "migrate" to another part of the spectrum upon modification.

The discussion of phasor interpolation to this point has ignored one important factor: the interpolated function $\mathcal{E}(\omega)$ is seriously affected by the phase terms $\{\theta_\ell\}$. To see this, consider the case when $\theta_\ell = 0$ for all ℓ; in this case, $\mathcal{E}(\omega)$ is simply a straightforward interpolation of the amplitude parameters. However, if every other phase term is π instead, $\mathcal{E}(\omega)$ interpolates adjacent amplitude parameters with opposite signs, resulting in a radically different set of modified amplitude parameters. It is therefore reasonable to formulate phasor interpolation such that the effects of phase on the modified amplitudes is minimized.

As mentioned above, when the phase terms are all close to zero, phasor interpolation approximates amplitude interpolation. Furthermore, examining Equation 4.35 reveals that when the phase terms are all close to π, phasor interpolation is approximately interpolation of amplitudes with a sign change, and that deviation from either of these conditions results in undesirable nonlinear amplitude interpolation. As described in the previous section, pitch onset time estimation is designed such that the "time-shifted" phase parameters $\{\psi_\ell(\tau_p)\}$ are close to zero or π; therefore, given the pitch onset time and $\{\psi_\ell(\tau_p)\}$, the phasor interpolation procedure outlined above may be performed using the "maximally coherent" phases instead of $\{\theta_\ell\}$, yielding the modified amplitude parameters $\{\hat{b}_\ell\}$ and interpolated phase parameters $\{\hat{\psi}_\ell(\tau_p)\}$.

Referring to Equation 4.29, the k-th modified excitation contribution should have a pulse located at $n = \tau_p^k/\beta_k - \delta^k$; however, using the interpolated phase parameters $\{\psi_\ell(\tau_p)\}$ in Equation 4.34 produces a modified excitation contribution with a pulse location of $n = -\delta^k$. Therefore, the phase terms must be adjusted such that a time shift of τ_p^k/β_k samples is imparted to $\hat{e}_{\beta_k}[n]$, yielding

$$\hat{\theta}_\ell^k = \hat{\psi}_\ell^k(\tau_p^k) - \ell\omega_o^k\tau_p^k. \tag{4.37}$$

At this point all that remains is to specify appropriate differential frequency terms in the equation for $\hat{e}_{\beta_k}[n]$. Although this task is somewhat arbitrary, by referring to the complex representation of $\tilde{s}^k[n]$ in Equation 4.13 it is reasonable to expect that the differential frequency terms may be interpolated uniformly in a manner similar to phasor interpolation, yielding

$$\hat{\Delta}_\ell = \sum_{i=0}^{J} \Delta_i I(\beta\ell\omega_o - i\omega_o). \tag{4.38}$$

This interpolation has the effect that the modified differential frequencies follow the same trend in the frequency domain as the unmodified differentials, which is important both in preventing migration of noise effects and in modifying speech segments which have a noise-like structure in certain portions of the spectrum [68].

Given the amplitude, phase and differential frequency parameters of the modified residual, the specification of a synthetic contribution to pitch-scale modified speech is completed by reintroducing the effects of the spectral envelope to the amplitude and phase parameters at the modified frequencies $\hat{\omega}_\ell^k = \beta\ell\omega_o^k + \hat{\Delta}_\ell^k$:

$$\begin{aligned} \hat{A}_\ell^k &= \hat{b}_\ell^k |H^k(e^{j\hat{\omega}_\ell^k})| \\ \hat{\phi}_\ell^k &= \hat{\theta}_\ell^k + \angle H^k(e^{j\hat{\omega}_\ell^k}) \end{aligned} \tag{4.39}$$

The parameter set thus determined may now be used in the modification synthesis of Equations 4.14 and 4.15, using the same time shift parameters as calculated in Equation 4.31.

CHAPTER 5

Computational Considerations

At first glance the analysis-by-synthesis algorithm described in Section 3.3 appears to involve a great deal of computation, since in each analysis frame five inner product expressions (Equation 3.27) must be evaluated a total of $M/2+1$ times for each of J sinusoidal components. Furthermore, direct evaluation of the overlap-add synthesis expressions (Equations 3.4 and 3.6) requires the direct computation of J sinusoids at $2N_s+1$ points in each synthesis frame, representing a prohibitive computational load for many real-time applications.

However, making use of the frequency-domain dualities derived in Section 3.3.2 leads to a formulation of analysis-by-synthesis which operates entirely in terms of discrete Fourier transforms. This frequency-domain formulation of analysis-by-synthesis may then be implemented using the FFT algorithm, significantly improving its computational efficiency. In addition, since constant-amplitude, linear-phase sinusoids are used in overlap-add synthesis, the *inverse FFT* (IFFT) algorithm may be used there as well. This chapter discusses techniques for exploiting these observations.

5.1 Use of the FFT in Analysis-by-Synthesis

The M-point DFT of an M-point sequence $x[n]$ is defined by

$$X[m] \triangleq \sum_{n=0}^{M-1} x[n] W_M^{mn}, \quad 0 \le m < M, \tag{5.1}$$

where

$$W_M^{mn} = e^{-j(2\pi/M)mn}; \tag{5.2}$$

the original sequence is recoverable from $X[m]$ using the *inverse DFT* formula

$$x[n] = \frac{1}{M}\sum_{m=0}^{M-1} X[m]W_M^{-mn}, \quad 0 \le m < M. \tag{5.3}$$

Comparing equation 5.1 with Equation 3.40, if $x[n]$ is nonzero only on the interval $[0, M-1]$, it is easily seen that

$$X[m] \equiv X(e^{j\omega})\Big|_{\omega=(2\pi/M)m}; \tag{5.4}$$

in other words, the DFT of a sequence with support limited as above is obtained by sampling the DTFT of the same sequence at M equally spaced points in the frequency range $0 \le \omega \le 2\pi$ [69]. For the purposes of analysis-by-synthesis, the M-point DFT's of $e_{\ell-1}[n]g[n]$ and $g^2[n]$ may be expressed as

$$\begin{aligned} EG_{\ell-1}[m] &= \sum_{n=-N_a}^{N_a} e_{\ell-1}[n]g[n]W_M^{mn} \\ GG[m] &= \sum_{n=-N_a}^{N_a} g^2[n]W_M^{mn}. \end{aligned} \tag{5.5}$$

Noting that $W_M^{-m(n+M)} = W_M^{mn}$, these DFT's may be cast in the form of Equation 5.1 (provided that $M > 2N_a$) by adding M to the negative summation index values and zero-padding the unused index values.

Recall now the relations derived in Section 3.3.2 between the inner product expressions calculated in analysis-by-synthesis and the DTFT's $GG(e^{j\omega})$ and $EG_{\ell-1}(e^{j\omega})$. From Equation 3.49,

$$\gamma_{11}^{\ell} = \frac{1}{2}\Re e\{GG(e^{j0}) + GG(e^{j2\omega_\ell})\}. \tag{5.6}$$

Clearly, from Equation 5.4 for the case of $\omega_\ell = \omega_c[i] = 2i\pi/M$, γ_{11}^{ℓ} is given by

$$\gamma_{11}^{\ell} = \frac{1}{2}\Re e\{GG[0] + GG[2i]\}. \tag{5.7}$$

Similarly, expressions for γ_{12}^{ℓ} and γ_{22}^{ℓ} can also be derived:

$$\begin{aligned} \gamma_{12}^{\ell} &= -\frac{1}{2}\Im m\{GG[2i]\} \\ \gamma_{22}^{\ell} &= \frac{1}{2}\Re e\{GG[0] - GG[2i]\}. \end{aligned} \tag{5.8}$$

Examining these relations, it is seen that the first three parameters are determined from the stored values of a single DFT which need only be calculated once per analysis frame. This DFT may be computed via the FFT algorithm using approximately $M \log_2 M$ complex multiplications and additions, yielding dramatic savings in computation over direct evaluation of the inner product terms.

Similar expressions for ψ_1^ℓ and ψ_2^ℓ are derived directly from Equation 3.49:

$$\psi_1^\ell = \Re e\{EG_{\ell-1}[i]\} \tag{5.9}$$

and

$$\psi_2^\ell = -\Im m\{EG_{\ell-1}[i]\}. \tag{5.10}$$

These parameters are thus expressed in terms of the stored values of $EG_{\ell-1}[m]$. Of course, since $e_{\ell-1}[n]$ changes for each new component added to the approximation, this DFT must be computed J times per frame. In order to reduce the amount of computation further, the relations derived in Section 3.3.2 may be used to update $EG_{\ell-1}[m]$.

Combining the results of Equations 3.43 and 3.45, the updated "error spectrum" $EG_\ell(e^{j\omega})$ is given by

$$EG_\ell(e^{j\omega}) = EG_{\ell-1}(e^{j\omega}) - \frac{1}{2}A_\ell e^{j\phi_\ell} GG(e^{j(\omega-\omega_\ell)}) - \frac{1}{2}A_\ell e^{-j\phi_\ell} GG(e^{j(\omega+\omega_\ell)}). \tag{5.11}$$

Making use of Equation 5.4, and recalling that $\omega_\ell = 2\pi i_\ell / M$, the updated error DFT $EG_\ell[m]$ is written as

$$EG_\ell[m] = EG_{\ell-1}[m] - \frac{1}{2}A_\ell e^{j\phi_\ell} GG[((m-i_\ell))_M] - \frac{1}{2}A_\ell e^{-j\phi_\ell} GG[((m+i_\ell))_M], \tag{5.12}$$

where $((\cdot))_M$ denotes the "modulo M" operator. $EG_\ell[m]$ can therefore be expressed as a simple linear combination of $EG_{\ell-1}[m]$ and circularly shifted versions of $GG[m]$. This method of updating $EG_\ell[m]$ is not only more elegant than that of subtracting successive components from $e_\ell[n]$ and recalculating the DFT, it also represents a considerable improvement in computational efficiency over the direct method. This is

because the component $\hat{x}_\ell[n]$ does not have to be evaluated after calculating its parameters, and because the FFT algorithm only has to be used once per analysis frame to calculate $EG_0[m]$; in fact, the only computation required to update $EG_\ell[m]$ is the approximately M additions and multiplications needed to implement Equation 5.12.

5.2 Use of the IFFT in Overlap-Add Synthesis

Referring to Equation 3.4 and using the inverse DFT formula of Equation 5.3, the expression for $\tilde{s}^k[n]$ may be written as

$$\begin{aligned} \tilde{s}^k[n] &= \sum_{\ell=1}^{J} A_\ell \cos(\omega_\ell n + \phi_\ell) \\ &= \Re e\left\{\frac{1}{M}\sum_{\ell=1}^{J} M A_\ell e^{j\phi_\ell} W_M^{-i_\ell n}\right\}. \end{aligned} \tag{5.13}$$

From this we see that $\tilde{s}^k[n]$ may be calculated by constructing an M-point sequence in m with values of $MA_\ell e^{j\phi_\ell}$ at $m = i_\ell$ and zero otherwise, then taking the real part of the inverse DFT of this sequence. This establishes a basis for using the IFFT algorithm to perform synthesis.

According to Equation 4.15, in the presence of time- and frequency-scale modification a synthetic contribution is given by

$$\tilde{s}^k_{\rho_k,\beta_k}[n] = \sum_{\ell=0}^{J[k]} A_\ell^k \cos(\hat{\omega}_\ell^k n + \zeta_\ell^k), \tag{5.14}$$

where

$$\hat{\omega}_\ell^k = \beta_k \ell \omega_o^k + \Delta_\ell^k/\rho_k, \tag{5.15}$$

$$\zeta_\ell^k = \phi_\ell^k + \beta_k \ell \omega_o^k \delta^k. \tag{5.16}$$

Except for the case when $\beta_k = \rho_k = 1$, the modified frequency terms of $\tilde{s}^k_{\rho_k,\beta_k}[n]$ no longer fall at multiples of $2\pi/M$; however, the IFFT algorithm may still be used to accurately represent $\tilde{s}^k_{\rho_k,\beta_k}[n]$. Ignoring frame notation, this is accomplished by

calculating DFT indices whose corresponding frequencies are adjacent to $\hat{\omega}_\ell$:

$$i_{1,\ell} = \left\lfloor \frac{\hat{\omega}_\ell \hat{M}_k}{2\pi} \right\rfloor \tag{5.17}$$

$$i_{2,\ell} = i_{1,\ell} + 1. \tag{5.18}$$

Recalling the discussion of approximation accuracy relative to frame length at the end of Section 3.3, it is important to adapt $\hat{M}_k$, the length of the DFT used in modification synthesis, so that consistent accuracy (as well as consistent computation) is achieved over the varying frame lengths required in time-scale modification. To this end, $\hat{M}_k$ may be set to

$$\hat{M}_k = \nu_s \rho_k N_s;$$

experiments with audio waveforms sampled at 8 kHz and 16 kHz indicate that a value of $\nu_s = 5$ is sufficient to guarantee high-quality synthesis over a wide range of modifications.

At this point, each component of $\tilde{s}^k_{\rho_k,\beta_k}[n]$ is approximated using two components with frequencies $\hat{\omega}_{1,\ell} = 2\pi i_{1,\ell}/\hat{M}_k$ and $\hat{\omega}_{2,\ell} = 2\pi i_{2,\ell}/\hat{M}_k$ in the following manner: Given a single sinusoidal component with an unconstrained frequency $\hat{\omega}_\ell$ of the form

$$c_\ell[n] = A_\ell \cos(\hat{\omega}_\ell n + \zeta_\ell) = \hat{a}_\ell \cos \hat{\omega}_\ell n + \hat{b}_\ell \sin \hat{\omega}_\ell n, \tag{5.19}$$

two sinusoids with constrained frequencies are added together to form an approximation to $c_\ell[n]$:

$$\begin{aligned} \tilde{c}_\ell[n] &= A_{1,\ell} \cos(\hat{\omega}_{1,\ell} n + \zeta_{1,\ell}) + A_{2,\ell} \cos(\hat{\omega}_{2,\ell} n + \zeta_{2,\ell}) \\ &= a_{1,\ell} \cos \hat{\omega}_{1,\ell} n + b_{1,\ell} \sin \hat{\omega}_{1,\ell} n + a_{2,\ell} \cos \hat{\omega}_{2,\ell} n + b_{2,\ell} \sin \hat{\omega}_{2,\ell} n. \end{aligned} \tag{5.20}$$

Letting $\hat{N}_s = \rho_k N_s$ and using the squared error norm

$$E_\ell = \sum_{n=-\hat{N}_s}^{\hat{N}_s} \{c_\ell[n] - \tilde{c}_\ell[n]\}^2, \tag{5.21}$$

minimization of E_ℓ in terms of the coefficients of $\tilde{c}_\ell[n]$ leads to the conditions

$$\frac{\partial E_\ell}{\partial a_{1,\ell}} = \frac{\partial E_\ell}{\partial a_{2,\ell}} = \frac{\partial E_\ell}{\partial b_{1,\ell}} = \frac{\partial E_\ell}{\partial b_{2,\ell}} = 0. \tag{5.22}$$

Expanding the first condition using Equation 5.20 yields

$$\sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \tilde{c}_\ell[n] \cos \hat{\omega}_{1,\ell} n = \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} c_\ell[n] \cos \hat{\omega}_{1,\ell} n. \tag{5.23}$$

Equations 5.19 and 5.20 may be substituted into this equation; however, noting that

$$\sum_{n=-N}^{N} \cos \alpha n \sin \beta n = 0$$

for all α, β and N, the resulting expression simplifies to

$$a_{1,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \cos^2 \hat{\omega}_{1,\ell} n + a_{2,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \cos \hat{\omega}_{1,\ell} n \cos \hat{\omega}_{2,\ell} n = \hat{a}_\ell \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \cos \hat{\omega}_\ell n \cos \hat{\omega}_{1,\ell} n. \tag{5.24}$$

Similarly, the other conditions of Equation 5.22 are given by the equations

$$a_{1,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \cos \hat{\omega}_{1,\ell} n \cos \hat{\omega}_{2,\ell} n + a_{2,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \cos^2 \hat{\omega}_{2,\ell} n = \hat{a}_\ell \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \cos \hat{\omega}_\ell n \cos \hat{\omega}_{2,\ell} n, \tag{5.25}$$

$$b_{1,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \sin^2 \hat{\omega}_{1,\ell} n + b_{2,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \sin \hat{\omega}_{1,\ell} n \sin \hat{\omega}_{2,\ell} n = \hat{b}_\ell \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \sin \hat{\omega}_\ell n \sin \hat{\omega}_{1,\ell} n, \tag{5.26}$$

and

$$b_{1,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \sin \hat{\omega}_{1,\ell} n \sin \hat{\omega}_{2,\ell} n + b_{2,\ell} \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \sin^2 \hat{\omega}_{2,\ell} n = \hat{b}_\ell \sum_{n=-\hat{N}_\ell}^{\hat{N}_\ell} \sin \hat{\omega}_\ell n \sin \hat{\omega}_{2,\ell} n. \tag{5.27}$$

Equations 5.24 and 5.25 form a pair of normal equations in the form of Equation 3.26 which may be solved using the formulas of Equation 3.28 for $a_{1,\ell}$ and $a_{2,\ell}$; likewise, Equations 5.26 and 5.27 are a second, independent pair of normal equations which yield $b_{1,\ell}$ and $b_{2,\ell}$.

Since the inner product terms in Equations 5.24–5.27 are given in terms of fixed functional forms, they may be calculated in closed-form using the relations

$$\sum_{n=-N}^{N} \cos \alpha n \cos \beta n = \frac{1}{2} F_N(\alpha - \beta) + \frac{1}{2} F_N(\alpha + \beta) \tag{5.28}$$

$$\sum_{n=-N}^{N} \sin \alpha n \sin \beta n = F_N(\alpha - \beta) - \sum_{n=-N}^{N} \cos \alpha n \cos \beta n, \tag{5.29}$$

where the function $F_N(\omega)$, defined as

$$F_N(\omega) \triangleq \frac{\sin(2N+1)\omega/2}{\sin \omega/2}$$

may be precalculated and used as required. Given parameters determined from the two sets of normal equations, the amplitude and phase parameters of $\tilde{c}_\ell[n]$ are derived using Equation 3.30. These parameters can then be assigned to the $\hat{M}_k$-point sequence $\hat{Z}[m]$ as described previously at index values $i_{1,\ell}$ and $i_{2,\ell}$. The inverse DFT of $\hat{Z}[m]$ may then be calculated by the inverse FFT algorithm using on the order of $\hat{M}_k \log_2 \hat{M}_k$ complex multiplications and additions.

5.3 Computational Comparisons of ABS/OLA and Sine-wave Transform Systems

In order to gauge the effectiveness of the computational enhancements described in this chapter within a meaningful context, it is useful to compare the resulting computational load of the ABS/OLA system with that of the Sine-wave Transform System of McAulay and Quatieri. The STS serves as a useful benchmark for comparison, not only because of its similarity to the ABS/OLA system, but also because it has been established as implementable in real time. These computational comparisons will not be exhaustive, but will instead focus on differences in the implementations of the two systems. The parameters used will correspond to those defined in this chapter and in Chapters 3 and 4.

5.3.1 Analysis Techniques

For each analysis frame, the peak-picking procedure used in the STS requires computation of a single M-point DFT of windowed signal data, using $M \log_2 M$ multiplications and additions. Establishing the locations of all spectral peaks requires another M multiplications and $M/2$ additions, and determining the J most significant peaks

requires an additional $2J^2$ multiplications and J^2 additions. Thus the total number of multiplications and additions needed per frame in peak-picking analysis is

$$M \log_2 M + M + 2J^2$$

and

$$M \log_2 M + M/2 + J^2,$$

respectively.

By contrast, the analysis-by-synthesis procedure of the ABS/OLA system requires computation of two M-point DFT's $EG_0[m]$ and $GG[m]$. Instead of peak-picking, an exhaustive frequency search procedure is used to determine the frequency of each component frequency ω_j^k, requiring calculation of $M/2$ sets of parameters and corresponding error terms per component; referring to Equation 3.28 and Equations 5.6–5.8, some parameters must be calculated only once per frame, resulting in a total of $3M/2$ additions and M multiplications.

Other parameters must be calculated for each component, requiring a total of $4JM$ multiplications and $3JM/2$ additions per analysis frame. Finally, as discussed above, updating the error DFT $EG_\ell[m]$ after determining the parameters of each component involves a total of JM multiplications and additions per analysis frame. The total computation required to implement these operations in analysis-by-synthesis is then

$$2M \log_2 M + M + 5JM$$

multiplications and

$$2M \log_2 M + 3M/2 + 5JM/2$$

additions.

Comparing the computational counts for the two analysis techniques, analysis-by-synthesis requires a surplus of $2M \log_2 M$ complex operations over peak-picking analysis due to calculation of an additional DFT, and approximately $7.5JM$ real

floating-point operations due to computation of least-squares parameters and the frequency search procedure. To get a feel for the magnitude of these surpluses, consider a typical case of analysis for $J = 30$ sinusoids using an $M = 512$-point DFT: the additional FFT requires 9 000 extra operations per analysis frame, while the search procedure contributes to a total surplus of 115 000 operations per frame; for a frame length of $N_s = 80$ samples[1], analysis-by-synthesis adds 1 400 floating point operations per sample to the computational load of analysis.

While the amount of computation of analysis-by-synthesis as implemented in this chapter is considerably higher than that required for peak-picking, it is worth reiterating that the comparison given above does not completely account for overhead computation involved in both algorithms, thus in the context of a full analysis/synthesis system the two algorithms are much closer in performance than it might appear; in actual implementations on general-purpose minicomputers and workstations[2], analysis-by-synthesis runs approximately fifty percent slower than peak-picking. For certain applications, however, the computational burden of analysis-by-synthesis as described in the thesis would be prohibitive; fortunately, in such circumstances it is possible to adjust analysis-by-synthesis to be much more efficient than the fully developed algorithm, with only slight decreases in performance.

For instance, while the overlap-add model defined in Equation 3.2 uses an envelope sequence $\sigma[n]$ to improve the model's performance in non-stationary portions of audio signals, this sequence is not required for the model to function. By assuming the steady-state condition where $\sigma[n] \equiv 1$ as described in Section 3.3.2, several computational advantages are gained: First, the computation required to estimate the envelope sequence as described in Section 3.2 is eliminated, saving $2I$ multiplications and I additions per sample. Second, as noted in Section 3.3.2, the DFT $GG[m]$ required to perform analysis-by-synthesis no longer varies from frame to frame, im-

[1] 10 msec at $F_s = 8000$ Hz.

[2] Multiflow Trace and Sun Sparcstation 2 computers.

plying that it may be stored once and read from memory as needed in each frame. Likewise, the inner product terms depending on $GG[m]$ may also be stored. As a result, analysis-by-synthesis without the use of $\sigma[n]$ requires

$$M \log_2 M + 5JM$$

multiplications and

$$M \log_2 M + 5JM/2$$

additions.

Unfortunately, eliminating the envelope sequence in analysis-by-synthesis reduces the overall computational load only slightly. In the example above, for instance, this version of analysis-by-synthesis still requires an extra 100 000 operations per frame over peak-picking. Examining the expressions for overall computation in analysis-by-synthesis given before, it is clear that the main computational bottleneck results from the $15JM/2$ operations involved in the exhaustive frequency search. In order to reduce this figure, consider the approximate relation for E_ℓ given in Equation 3.53, which is repeated here:

$$E'_\ell \approx E'_{\ell-1} - \frac{|EG_{\ell-1}(e^{j\omega_\ell})|^2}{\frac{1}{2}W_a(e^{j0})}. \tag{5.30}$$

This relation holds for all but very low or very high frequencies, where little perceptually important energy is found; thus, assuming equality in this relation has little impact on the overall performance of analysis-by-synthesis, and considerably simplifies the frequency search procedure. According to Equation 3.53, the optimal component frequency corresponds to the maximum of $|EG_{\ell-1}[m]|^2$. The overall computation required then reduces to

$$M \log_2 M + 6J + 2JM$$

multiplications and

$$M \log_2 M + 2J + 3JM/2$$

additions. Given the parameters used for comparison as above, this version of analysis-by-synthesis requires 50 000 operations per frame more than peak-picking analysis, a twofold decrease in the surplus of the fully developed analysis-by-synthesis algorithm.

5.3.2 Synthesis Techniques

As discussed in Section 1.3, the notion of performing synthesis using overlap-add techniques which incorporate the IFFT algorithm is not novel to the ABS/OLA system, and has been explored by McAulay and Quatieri for use in the Sine-wave Transform System [43]. As mentioned previously, their approach is based on matching components in adjacent analysis frames, estimating the parameters of a "midframe" synthetic contribution, then performing overlap-add synthesis at half the frame rate of analysis, using $M = 512$-point DFT's (for $F_s = 8000$ Hz) to generate synthetic contributions. One synthetic contribution is passed from the previous frame, and two new contributions must be calculated to generate the remainder of the frame. Thus, a total of 9200 complex multiplications and additions are required to generate a synthesis frame of $\hat{N}_s$ points, provided that the synthesis frame is less than 20 msec long.

Unlike the overlap-add synthesis strategy used in the Sine-wave Transform System, the ABS/OLA system requires two rather than three synthetic contributions to generate a synthesis frame; this is possible due to the refined modification model derived in Section 4.3.2. Synthesis accuracy is maintained in this model over variable synthesis frame lengths by keeping the DFT length proportional to $\rho_k N_s$. For the typical case used in the last section for comparison, the amount of computation required to generate a synthesis frame is approximately $400\rho_k \log_2 400\rho_k$ operations. On a per sample basis, the ABS/OLA system requires approximately

$$5 \log_2 400\rho_k$$

operations, while the Sine-wave Transform System requires $115/\rho_k$ operations. For a time scale factor of one, the ABS/OLA system uses approximately 45 complex

operations per sample, less than half the amount of computation used in the Sine-wave Transform System. The computational performance of the STS improves for higher time scale factors, becoming approximately equal to the ABS/OLA system for $\rho_k = 2$.

Besides the result that synthesis in the ABS/OLA system is generally more efficient than in the Sine-wave Transform System, two other important points are apparent from the above discussion: First, the computational complexity of STS synthesis is inversely proportional to ρ_k, while in the ABS/OLA system the computational rate is proportional to $\log_2 400\rho_k$, which is much more consistent over typical time scale factor values. Second, while the STS might appear to be more efficient than the ABS/OLA system for time scale factors greater than two, it is important to remember that in this example a value of $\rho_k = 2$ corresponds to a synthesis frame length of 20 msec, which is the maximum allowable frame length for the Sine-wave Transform System. Thus, in order to perform time-scale modification at higher factors with comparable quality it is necessary to use smaller values of N_s; doing so increases the analysis frame rate, which in turn increases the computational load of the analysis procedure. By contrast, the ABS/OLA system, which uses variable length DFT's in conjunction with a refined overlap-add modification model, is capable of performing time-scale modification over a broader range of scale factors with more consistent quality and rate of computation.

CHAPTER 6

Application of ABS/OLA System to Music Synthesis

The ABS/OLA system described in Chapters 3–5 has been designed specifically to perform speech analysis/synthesis and speech modifications by making use of the quasi-harmonic representation of Equation 4.4. The quasi-harmonic overlap-add sinusoidal model represents speech signals very well, since it captures the harmonic character of speech as well as time variations of pitch, voicing state and spectral content. However, these properties are not exclusive to speech signals; pitched musical instruments such as horns, woodwinds and stringed instruments produce quasi-harmonic signals as well. In fact, with the exception of pitch-scale modification, all of the modifications described in the preceding chapters may be performed on arbitrary pitched musical tones with equal effectiveness.

In several ways, music signals are easier to process than speech. For instance, the fundamental frequency of sustained pitched tones is approximately known *a priori*. In addition to eliminating the need for pitch tracking algorithms, this knowledge allows analysis and synthesis to be tuned for optimum performance at the given pitch. Another advantage to analyzing pitched musical tones is that except for well-defined *attack* and *release* portions,[1] musical tone signals are more stationary and easily represented with a quasi-harmonic model than speech signals; this makes their analysis simpler and modification more robust.

[1] *Attack* - the onset portion of a musical tone; *release* - the closing portion of a musical tone.

However, the quality requirements of music synthesis (which is often used in high-fidelity audio environments) are much more stringent than those of speech synthesis, where intelligibility is often of greater concern than fidelity. These requirements impact both on the accuracy needed in analysis and the bandwidth at which tones are synthesized. This chapter discusses the adaptations required to use the ABS/OLA system for high-fidelity music synthesis and modification, and begins with a background discussion of popular approaches to music synthesis using computers.

6.1 Digital Music Synthesis (DMS) Techniques

One of the earliest attempts to synthesize music digitally in the time-domain was made by Mathews in 1958 [70]. In his technique, known as *fixed-waveform synthesis* (FWS), N samples of one period of a periodic waveform (typically calculated from Fourier coefficients) are stored in a *wavetable* denoted by $x[n]$. The synthetic signal $s[n]$ is then generated by

$$s[n] = A[n]x[T[n]], \tag{6.1}$$

where $A[n]$ is a time-varying amplitude envelope. The *index* $T[n]$ is given by

$$T[n] = ((T[n-1] + Nf[n]/F_s))_N, \tag{6.2}$$

where $f[n]$ is the time-varying fundamental frequency in Hz. For a fixed value of $f[n]$, this *cyclic table lookup* procedure yields periodic replications of $x[n]$ multiplied by $A[n]$.

Unfortunately, synthetic music produced by FWS sounds very unnatural and "mechanical." The reason for this is its inability to capture the time variation of spectral content critical to the *timbre* or tone quality of a given instrument. Nevertheless, this technique is very simple and fast and is often used when quality requirements are not high or when hardware is limited. Furthermore, wavetables remain the method of choice for producing waveforms in other DMS techniques.

Since the timbre of a musical tone depends on its behavior from beginning to end, an obvious extension of fixed-waveform synthesis is to store the entire tone in a wavetable and play it back on demand. This is the basis of *sampling*, one of the most popular DMS techniques in use today.

Some control over basic parameters such as the frequency- and time-scale of sampled tones is necessary for sampling to be useful. The simplest approach to time-scale modification of sampled tones is *looping*, whereby the steady-state portion of a tone is simply repeated to lengthen a tone or bypassed to shorten it. When combined with sampling rate changes, frequency-scale modification results. A refinement of looping is the *Synchronized Overlap-Add* method of Roucous and Wilgus [35], which performs time-scale modification by repeating or omitting short segments of the sampled tone while accounting for phase coherence. However, these techniques suffer from objectionable artifacts due to spectral discontinuities.

Sampling enjoys the advantage that sampled sounds can be played back with very high fidelity, and when combined with looping provides enough control for many musical applications. Also, sampling lends itself to very simple hardware implementations, explaining its popularity. However, a considerable drawback to sampling is that very little control beyond crude time- and frequency-scale modification is possible since there is no underlying model which produces the tone.

By way of contrast, *additive techniques* typically employ signal models which allow for flexible modification of analyzed tones. Perhaps the most well known additive technique is *sinusoidal additive synthesis* (SAS). Very similar to the ABS/OLA model, SAS is a time-frequency representation of musical signals. This characteristic is critical to its performance for, as previously mentioned, any synthesis technique must capture the time-variant spectral character of a musical tone in order to effectively reproduce its timbre.

Additive synthesis is described by the equation

$$s[n] = \sum_{k=0}^{K} A_k[n] \cos(\Theta_k[n]), \tag{6.3}$$

where the *control functions* $A_k[n]$ and $\Theta_k[n]$ are the time-varying amplitude and phase, respectively, of the k-th partial. $\Theta_k[n]$ is given recursively as

$$\Theta_k[n] = \Theta_k[n-1] + \omega_k[n], \tag{6.4}$$

where $\omega_k[n]$ represents the time-varying frequency of the k-th partial. Given these relations, it is possible to create a wide variety of musical sounds by specifying the amplitude and frequency control functions of each of the $K+1$ sinusoids.

In an early application of SAS, Jean-Claude Risset used time-dependent spectral analysis of trumpet tones to formulate a simple time-frequency model of that instrument. This enabled him to use SAS to produce realistic synthesized trumpet tones whose character could change with intensity to match the behavior of the natural instrument [71]. Unfortunately, such manual analysis procedures require a great deal of time, expertise and insight into the behavior of musical instruments. Furthermore, they are not amenable to analyzing more complicated instruments, especially the singing voice.

Were it always necessary to manually analyze tones and specify control functions, additive synthesis would not be considered a useful technique. However, one of the greatest advantages of SAS is that it is possible to analyze sampled tones automatically using the DSTFT, just as with time-frequency approaches to speech processing. The most popular approach to analysis in additive synthesis uses the digital phase vocoder to determine appropriate amplitude and frequency control functions $A_k[n]$ and $\omega_k[n]$. The ability to analyze real musical sounds, combined with the functional form of the model parameters, makes additive synthesis capable of both reproducing musical tones and performing a wide variety of useful modifications.

As mentioned before, though, the amount of computation required to synthesize a sum of sinusoids with arbitrary amplitude and phase functions is formidable, since wavetables must be used to generate individual components. For this reason it is necessary in real-time applications to limit the number of partials that can be used, often greatly reducing the fidelity of the synthetic tone. Furthermore, since the

computational load is so high, SAS cannot be implemented using inexpensive hardware. As in the case of speech processing, computation can be greatly enhanced by implementing the DSTFT using the FFT algorithm, but at the cost of modification artifacts [31, 32, 33].

Of course, sinusoids are not the only functions which can be added together to produce complex waveforms. For certain applications other basic functions are preferable. One such application is the synthesis of sung vowels. It is well known in speech processing that vowels have a *formant structure* which can be modeled using an all-pole filter excited by a pulse train at the desired pitch period. By partial fraction expansion this filter may be represented as a bank of parallel filters of low order, each corresponding to a single formant, which are excited by the pulse train and whose outputs are summed to produce the synthetic vowel.

Formant synthesis techniques generalize this process by using waveforms rather than filters in the parallel structure to represent formant behavior. Synthetic vowels are thus produced by

$$s[n] = A[n] \sum_k \{ \sum_{i=1}^{I} s_i[n - kN] \}, \tag{6.5}$$

where $A[n]$ is the amplitude envelope, N is the pitch period, I is the number of formants (typically a small number) and $s_i[n]$ is the waveform corresponding to the i-th formant. The remaining problem is to specify the formant waveforms $s_i[n]$ for a given vowel sound.

One approach to formant synthesis is *voice simulation* (VOSIM) [72]. It uses formant waveforms composed of N concatenated raised-cosine pulses of duration T. The amplitude of the first pulse is one, and successive pulses are related to each other by a multiplicative factor b. By analyzing the Fourier transform of this signal, it is seen that T controls the formant frequency and b and N control the formant's "skirt width" and bandwidth respectively. The resulting signal is then scaled to account for the relative amplitude of the formant, yielding $s_i[n]$. The difficulty with VOSIM is that the formant waveform spectrum exhibits ripples, and the formant parameters

are considerably coupled, making the system difficult to use.

An alternative to the VOSIM approach is the *Time-Domain Formant-Wave-Function* (FWF) synthesis technique proposed by Rodet [73] which, by analogy to a second order filter excited by a more sophisticated pulse waveform, uses the causal formant waveform given by

$$s_i[n] = \begin{cases} \sin^2(\beta_i n/2)\epsilon^{-\alpha_i n}\cos(\omega_i n + \phi_i), & 0 \leq n \leq \pi/\beta_i \\ \epsilon^{-\alpha_i n}\cos(\omega_i n + \phi_i), & n > \pi/\beta_i. \end{cases} \tag{6.6}$$

Like VOSIM, the formant frequency, bandwidth and skirt width are controlled by ω_i, α_i and β_i respectively. Unlike VOSIM, however, these formant parameters are largely decoupled, and the formant spectrum is considerably smoother. Analysis of vowel sounds using FWF is performed manually, with the formant parameters adjusted to match the spectrum of the original. When factors such as loudness-dependent spectral tilt, vibrato and transitional behavior are taken into account, FWF synthesis of the singing voice is most impressive [74].

Since formant synthesis requires fewer wavetables than SAS, its computational requirements are considerably lower, making it an attractive alternative to the more general technique. Also, as the pitch of the synthetic sound is changed, its formant structure is unaltered, making formant synthesis useful for modeling sounds such as speech which obey this property. However, formant synthesis is applicable only to sounds which possess a formant structure, and the lack of an automatic analysis technique limits its usefulness.

While time-domain techniques are very computationally efficient, they are incapable of providing fine control over the timbres they produce or of creating new musical sounds. Additive techniques are capable of creating arbitrarily complex timbres, but at the expense of efficiency and ease of control. The class of DMS techniques known as *nonlinear techniques* use modulation and distortion to create complex sounds with a small parameter set and with very little computation.

The oldest and simplest synthesis technique using modulation principles is *ring*

modulation (RM). It is widely used in analog music synthesis and is still used for various purposes in digital music synthesis. The formula for RM is

$$s(t) = x(t)\cos 2\pi f_c t. \tag{6.7}$$

In communication parlance, this is simply double sideband suppressed carrier amplitude modulation. The difference between the communication application and this is that f_c is in the audio frequency range.

The frequency-domain effect of RM can be seen as follows: If $x(t)$ is a real periodic signal with fundamental frequency f_m expressed as

$$x(t) = \sum_{k=0}^{N} A_k \cos(2\pi k f_m t + \phi_k), \tag{6.8}$$

then the Fourier transform of $x(t)$ is simply

$$X(f) = \sum_{k=-N}^{N} \alpha_k \delta(f - k f_m), \tag{6.9}$$

where of course $\alpha_{-k} = \alpha_k^*$. The Fourier transform of $s(t)$ is then given by

$$S(f) = \frac{1}{2}[X(f + f_c) + X(f - f_c)] \tag{6.10}$$

$$= \frac{1}{2}[\sum_{k=-N}^{N} \alpha_k(\delta(f + f_c - k f_m) + \delta(f - f_c - k f_m))]. \tag{6.11}$$

This is illustrated in Figure 6.1.

As seen from this figure, the relation between f_m and f_c is critical in determining the frequencies of the components of $s(t)$ and the shape of the resulting spectrum. In fact, very general statements can be made of any spectrum in the form of Equation 6.11 in terms of the value of f_c/f_m, known as the *C:M ratio* [44]. Depending on the *C:M* ratio, $s(t)$ will correspond to a harmonic signal as in Figure 6.1(a), a harmonic signal with certain harmonics missing as in Figure 6.1(b), or an inharmonic signal as in Figure 6.1(c). These characteristics are clearly useful in designing sounds to have desired properties, and in fact more general organizational techniques have been applied to classify *C:M* ratios for compositional purposes [75].

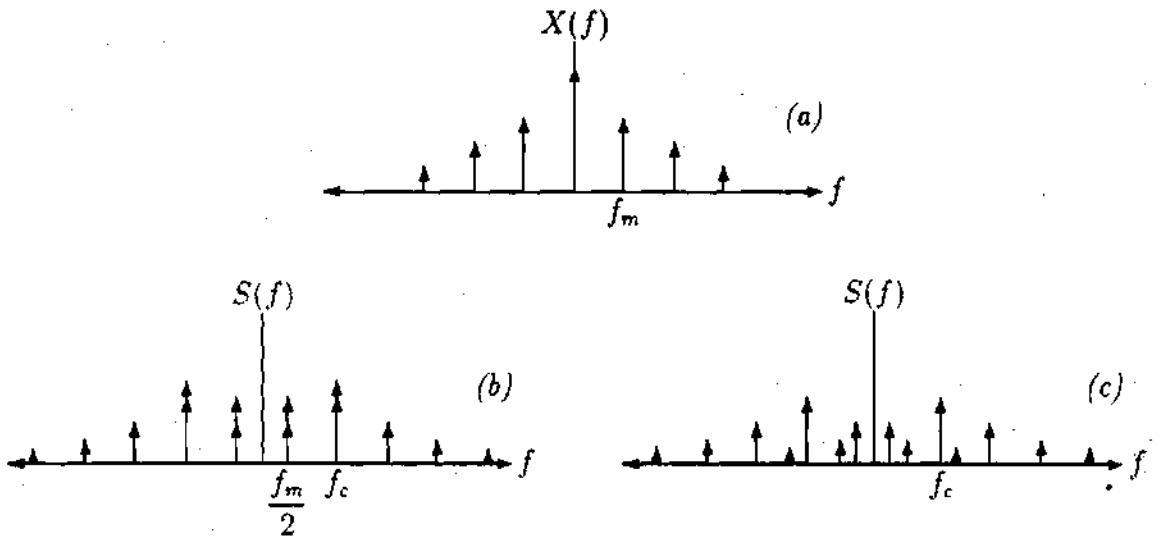

Figure 6.1: Illustration of spectra resulting from ring modulation: *(a)* Spectrum of periodic modulating signal $x(t)$ for $N = 3$; *(b)* Spectrum of $s(t)$ resulting when $f_c = \frac{3}{2}f_m$; *(c)* Inharmonic spectrum resulting when $f_c = \sqrt{2}f_m$.

Ring modulation is nearly as simple as fixed-waveform synthesis; it can be implemented using a multiplier and two wavetables, one loaded with a period of $x(t)$ and the other loaded with a cosine function. Since RM is capable of producing a variety of spectra at a variety of fundamental frequencies, it is considerably more useful than FWS. Its major drawback is that it is incapable of producing time-varying spectra, hence it is rarely used in isolation. Nevertheless, its ability to produce inharmonic spectra from harmonic spectra makes it very useful when combined with other techniques.

One of the most widely used techniques for digital music synthesis is *frequency modulation synthesis* (FM), introduced by John Chowning in 1973 [44]. This technique has its roots in communication theory, with the observation that when even a simple signal frequency modulates a carrier, the result can be a signal with a much greater bandwidth than the modulator and whose spectrum is very complex.

Chowning's formulation, known as *simple FM*, is given by the equation

$$s(t) = A(t)\sin(2\pi ct + I(t)\sin(2\pi mt)), \tag{6.12}$$

where $A(t)$ is an envelope signal which controls loudness, c is the *carrier frequency*, m is the *modulating frequency*, and $I(t)$ is the *index of modulation*. For fixed values of $A(t)$ and $I(t)$, $s(t)$ may be written as [76]

$$s(t) = A\sum_{k=-\infty}^{\infty} J_k(I)\sin(2\pi(c+km)t), \tag{6.13}$$

where $J_k(I)$ are k-th order Bessel functions of the first kind. Clearly the Fourier transform of $s(t)$ may be expressed in the form of Equation 6.11, hence the statements made before concerning the *C:M* ratio apply to signals produced by simple FM as well.

Examples of the one-sided spectra produced by simple FM for several values of I are shown in Figure 6.2. An important aspect of the behavior of FM is clear from this: As the modulation index increases, harmonics farther from the carrier frequency increase in magnitude. In perceptual terms, the resulting signal becomes *brighter* as

the index increases. This behavior, coupled with the ability to vary the modulation index with time, explains much of the power of FM to mimic the timbre of natural instruments.

However, increasing bandwidth implies that the modulation index I must be limited to prevent aliasing distortion. Also, the relationship of the modulation index to the spectral content of $s(t)$ is very complex and counterintuitive, meaning that in order to achieve specific results using FM it is often necessary to resort to trial-and-error. Furthermore, this complex linkage makes automatic analysis of sampled tones using simple FM impossible.

Even with these disadvantages is mind, FM has one compelling advantage: It is *very* simple to implement. Simple FM can be implemented using two wavetables, or roughly twice the computational load of fixed-waveform synthesis. Considering that FM is capable of producing harmonic, inharmonic and time-varying spectra and produces reasonable facsimiles of most instruments known, twice the computational level of FWS is a small price indeed.

The ability to control output spectra is important for many musical applications and is very difficult with FM Synthesis. A DMS technique that enjoys nearly the same computational load as FM and whose spectrum is controllable is *waveshaping synthesis*. As any audiophile knows, if a sinusoid of a given frequency and amplitude is fed through a nonlinear amplifier, the output signal is harmonic with the same frequency but contains components at multiples of the original fundamental. Waveshaping makes deliberate use of such nonlinear functions to produce *harmonic distortion*, and hence complex spectra, from simple sinusoidal inputs.

The general formulation of waveshaping is [77]

$$s(t) = f(a(t)\cos 2\pi f_m t), \tag{6.14}$$

where $f(x)$ is the *shaping function* and $a(t)$ is the *index*. While any arbitrary $f(x)$ will produce a complex spectrum, in order to analyze and design such spectra it is

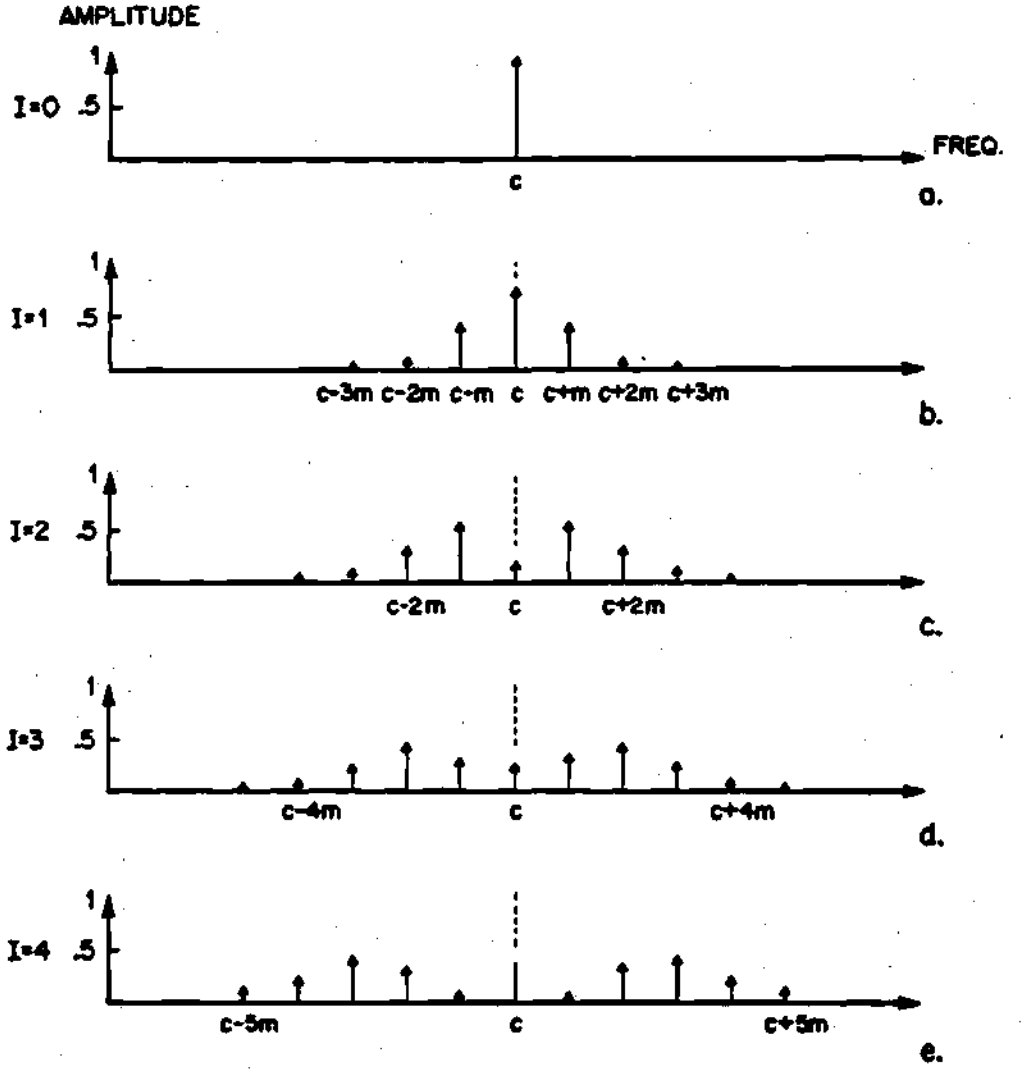

Figure 6.2: One-sided spectra resulting from simple FM for several values of I.

necessary to consider simpler *polynomial* shaping functions of the form

$$f(x) = \sum_{i=0}^{I} d_i x^i. \tag{6.15}$$

Polynomial shaping functions have the distinct advantage that their output spectra are bandlimited to If_m.

By expressing this function as a sum of *Chebychev polynomials* of the first kind, shaping functions can be designed to produce spectra with specific component magnitudes. To see this, recall that a fundamental relation of Chebychev polynomials is $T_i(\cos\theta) = \cos i\theta$ and hence

$$f(\cos 2\pi f_m t) = \sum_{i=0}^{I} h_i T_i(\cos 2\pi f_m t) = \sum_{i=0}^{I} h_i \cos 2\pi i f_m t. \tag{6.16}$$

By specifying the desired coefficients h_i as a vector, it is then possible to derive the polynomial coefficients by multiplying the vector with a transition matrix [78].

Note that Equation 6.16 only holds for an index value of one. As the index value is changed the component magnitudes vary in complex ways, making precise spectral control difficult. However, given the polynomial coefficients and an index value the resulting spectrum can be analyzed by recalculating the coefficients as $d_i(a) = a^i d_i$ and multiplying by the inverse of the aforementioned transition matrix.

One problem with waveshaping is that since the index is the amplitude of the input sinusoid, the overall loudness of the synthetic tone depends on the index. If the loudness is supposed to remain constant, this means that some form of *normalization* is necessary. Also, waveshaping by itself is capable of producing only harmonic tones, requiring that it be combined with ring modulation to produce more complex frequency-domain behavior. However, waveshaping is implemented with roughly the same complexity as RM by itself and is capable of producing time-varying, bandlimited spectra, making it a very popular synthesis technique.

While waveshaping synthesis provides more spectral control than FM, it is still difficult to predict the exact spectral behavior as the index is changed. A synthesis

technique which overcomes this problem is *summation formula synthesis* (SFS), introduced by Moorer [79]. SFS is based on the realization that for well-defined sets of magnitudes and frequencies, additive combinations of sinusoids can be expressed in closed-form using relatively simple formulas. For example,

$$\begin{aligned} s(t) &= \sum_{k=0}^{N} a^k \sin(\theta + k\beta) \\ &= \frac{\sin\theta - a\sin(\theta-\beta) - a^{N+1}[\sin\{\theta+(N+1)\beta\} - a\sin(\theta+N\beta)]}{1+a^2-2a\cos\beta}. \end{aligned}$$

When $\beta = 2\pi f_m t$ the resulting signal is in the form of Equation 6.8 which can then be ring modulated to produce a spectrum such as in Equation 6.11. Also, the index a can change with time to produce dynamic spectra.

The key advantage of SFS is that variations of the index a produce predictable changes in the resulting spectrum. Since it yields bandlimited spectra, aliasing is not a problem. However, since the index affects energy as well as spectral shape, normalization is necessary. Furthermore, SFS requires a minimum of four wavetables, a divider and an exponentiator to implement, making it the most computationally intense nonlinear synthesis technique.

Digital filters have been commonly used in speech processing for years, but have only recently begun to play an important role in digital music synthesis. *Digital filter based techniques* represent one of the most promising research areas in computer music today. The simplest such technique is *digital subtractive synthesis* (DSS). The basic idea of DSS is to start with a harmonically rich excitation signal such as a square or sawtooth signal produced by an oscillator, then to filter the excitation to alter its spectral shape.

Digital subtractive synthesis is typically implemented using simple second order all-pole filters connected in cascade or in parallel. The control parameters for each filter are the center frequency and bandwidth of the filter formant, which are mapped to the filter coefficients. Thus by varying these parameters it is possible to change the formant structure of the overall filter. The excitation signal can be random or

deterministic or a combination of the two, but is usually broadband. This technique is not very versatile, is difficult to control for many sections and requires power normalization. Nevertheless, it can be implemented inexpensively and is widely used when quality requirements are low.

A familiar theme to anyone working in speech processing is the use of digital filter structures to model the vocal production process for speech synthesis. The use of digital filters to model sound production in musical instruments has become a topic of great interest in the computer music community recently. Musical instruments are more amenable to such modeling than speech, since many of the resonant structures of traditional instruments are inherently time-invariant and can be more closely approximated using linear models.

The process of *physical modeling* is also familiar to speech processing engineers. Starting with knowledge of the mechanical structure of an instrument and the laws of physics, a set of coupled nonlinear partial differential equations and boundary conditions describing the sound production process of the instrument is derived. After considerable simplification the equations are represented as a set of coupled linear difference equations which can be implemented in a digital filter structure. Such analyses have been performed for bowed-string instruments [80] as well as for more general classes of instruments [81].

By far the most successful application of physical modeling to date is the plucked string algorithm of Karplus and Strong [82]. Based on an analysis of the vibrational behavior of strings [83], this technique simulates string vibration using the remarkably simple relation

$$s(n) = \frac{\rho}{2}s(n-N) + \frac{\rho}{2}s(n-N-1) + x(n), \tag{6.17}$$

where $\rho \leq 1$ is the *decay factor*, N is approximately the pitch period of the sound and $x(n)$ is an N-sample noise burst which represents the string pluck. This is nothing more than an $(N+1)$-order recursive filter whose pole-zero plot is shown in Figure 6.3 ($N = 12$) and $\rho = 1$.

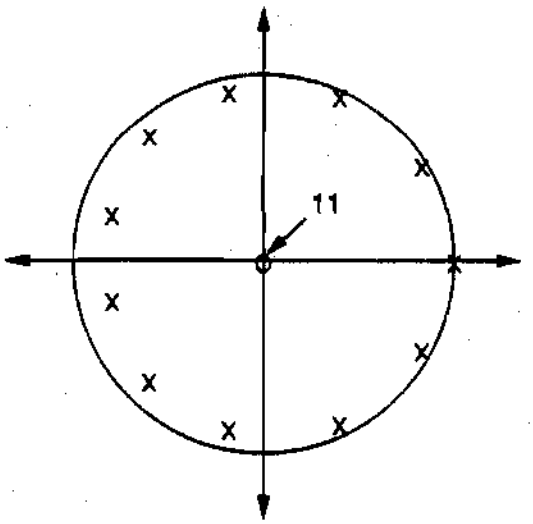

Figure 6.3: Pole-zero plot of Karplus-Strong filter for $N = 10$ and $\rho = 1$.

Despite its simplicity, the Karplus-Strong algorithm manages to capture many of the salient features of plucked string dynamics. Examining Figure 6.3, it can be seen that the pole locations lie further within the unit circle with increasing frequency. Also, the frequency locations of the poles are close to (but not exactly) multiples of $2\pi/(N + 1/2)$. Thus the impulse response is a sum of exponentially damped sinusoids that are approximate multiples of a fundamental frequency, and the high-frequency sinusoids decay faster than the low-frequency sinusoids. This is completely consistent with the observed behavior of plucked strings.

A variety of refinements have been made to the basic algorithm [84] which allow arbitrary tuning, simulate instrument characteristics such as string stiffness and sympathetic vibration, and provide for a variety of performance effects. While this algorithm does not accurately model the sound of existing stringed instruments it nonetheless sounds like a stringed instrument of some sort.

The advantage of physical modeling is that unlike other synthesis models, the instrument and performance parameters are directly accessible in the model and can be changed at will with predictable results. Furthermore, physical modeling makes it possible to extrapolate from given instruments to synthetic instruments which could not be built in the real world. Unfortunately, analyzing a given instrument's physical structure is a very difficult and not always successful process, and although recent advances in VLSI architectures have improved the situation, implementing high-order digital filters in real-time remains a difficult problem.

Based on the foregoing discussion, it is clear that a strong tradeoff exists: On the one hand, very computationally simple approaches to digital music synthesis either produce poor representations of natural musical instruments or allow minimal flexibility to modify the sounds produced. On the other hand, very powerful and flexible techniques which produce high-fidelity replicas of musical instruments are so computationally intense that their implementation is often prohibitively expensive. At the present time, there is a needed shortage of music synthesis techniques which

fall between the two extremes.

However, due to the similarity of speech and music signals, the ABS/OLA system is capable of analyzing pitched musical tones and synthesizing faithful replicas of the same. In addition, the refined quasi-harmonic model formulation allows modifications to be performed without the reverberant distortion associated with other time-frequency approaches to music synthesis. Furthermore, the computational shortcuts described in Chapter 5 allow the ABS/OLA system to be implemented much more simply than other approaches of similar quality and flexibility. The following section describes alterations of the ABS/OLA system made possible when analyzing and synthesizing pitched musical tones.

6.2 Design Details

Since the goal of sinusoidal modeling of musical tones is synthesis and modification, the quasi-harmonic model formulation of Equation 4.4, rewritten here for convenience, is used in music synthesis:

$$\tilde{s}^k[n] = \sum_{j=0}^{J[k]} A_j^k \cos((j\omega_o^k + \Delta_j^k)n + \phi_j^k). \tag{6.18}$$

As in the case of speech analysis, the fundamental frequency ω_o^k and the amplitude, frequency and phase parameters of $\tilde{s}^k[n]$ must be computed in each analysis frame. However, since an unambiguous initial estimate of ω_o^k is available *a priori*, the analysis-by-synthesis algorithm described in Section 3.3 may be modified both to simultaneously refine the initial estimate using calculated parameters and to ensure the resulting parameter set maintains the structure of Equation 6.18.

Harmonically-constrained analysis-by-synthesis begins with an initial fundamental frequency estimate $\omega_o^k = \omega_c' = 2\pi f_o'/F_s$, where f_o' is the nominal pitch frequency of the tone in Hz. As each candidate frequency $\omega_c[i]$ is tested to determine the ℓ-th component of $\tilde{x}[n]$, its harmonic number is calculated as the nearest integer to

$\omega_c[i]/\omega_o^k$. If this equals the harmonic number of any of the previous $\ell - 1$ components determined, the candidate is disqualified, thus insuring that only one component is associated with each harmonic number. As each new component is determined, the estimate of ω_o^k is updated according to Equation 4.6. This algorithm eliminates the need for further processing to resolve pitch ambiguity, considerably reducing the required computational overhead for analysis.

Given a tone's nominal pitch frequency, that knowledge may be used to optimize the performance of the analysis algorithm. Defining the nominal pitch period of a given musical signal in samples as $N_o' = F_s/f_o'$, note that the number of sinusoids in Equation 6.18 is approximately $N_o'/2$. Since each sinusoid has three parameters, there are approximately $\frac{3}{2}N_o'$ model parameters corresponding to each N_s samples of the original sequence. In order to maintain the same number of parameters as samples of the original sequence, and to make the number of model parameters independent of the pitch frequency, the synthesis frame length may thus be set to $N_s = \frac{3}{2}N_o'$. Furthermore, the analysis frame length $2N_a + 1$ should be at least $2N_o'$ in order to insure adequate spectral resolution in the analysis procedure, but should be as small as possible to avoid violating the assumption of stationarity. Letting $N_a = N_s$ provides an analysis frame length of $3N_o'$, which is sufficient.

Pitch information also allows for more specific design of the digital filter used to estimate $\sigma[n]$. Referring to Equation 3.10, the transfer function $F(e^{j\omega})$ corresponds approximately to a simple first-order lowpass filter section with unity DC gain, a 3-dB bandwidth controlled by λ, and a -6 dB/octave rolloff. Given a quasi-periodic sequence $s[n]$ with nominal fundamental frequency ω_o', $|s[n]|$ is also periodic with the same frequency. In order to capture the changes in energy of a tone as rapidly as possible without introducing ripple to $\sigma[n]$, the filter should attenuate all frequencies above ω_o' while passing information below ω_o'. A simple but effective constraint which achieves this goal is to adjust λ such that the half-power frequency of $F(e^{j\omega})$ occurs

at $\omega = \omega_o'$. Quantitatively, this is given by

$$|F(e^{j\omega_o'})|^2 = \frac{(1-\lambda)^2}{1+\lambda^2-2\lambda\cos\omega_o'} = \frac{1}{2}. \tag{6.19}$$

Solving this expression for λ leads to a quadratic equation with two real roots; however, only one root leads to a stable filter. This yields λ as a function of ω_o':

$$\lambda(\omega_o') = \xi - \sqrt{\xi^2 - 1}, \tag{6.20}$$

where $\xi = 2 - \cos\omega_o'$. The parameter $\lambda(\omega_o')$ has limiting values of 1 at $\omega_o' = 0$ and $3 - \sqrt{8}$ at $\omega_o' = \pi$ and decreases monotonically with increasing frequency. Since the attenuation at $\omega = \omega_o'$ is now known to be 3 dB, cascading I sections provides a minimum of $3I$ dB attenuation of signal components and $-6I$ dB/octave rolloff; I can therefore be adjusted to provide a desired amount of attenuation and rolloff. Experiments indicate that $I = 14$ (40 dB attenuation) is a minimum number of sections to use for most instrumental tones. The filter delay n_o is then calculated according to Equation 3.13, and is inversely proportional to the nominal fundamental frequency. Using $I = 14$ (40 dB attenuation), $n_o \approx 2N_o'$.

Referring to the discussion of speech modification in Section 4.3, one of the main advantages of the coherence preservation algorithm used to calculate time shifts δ^k and δ^{k+1} for speech modifications is that the algorithm is relatively insensitive to errors in fundamental frequency estimation resulting in an estimate which is the actual fundamental multiplied or divided by an integer factor. However, for the case of pitched musical tones, such considerations are irrelevant since the fundamental frequency is approximately known *a priori*. Therefore, a simpler constraint may be invoked to determine appropriate time shifts.

Specifically, denoting the phase terms of the sinusoids in Equation 4.15 by $\hat{\Phi}_j^k[n]$ and $\hat{\Phi}_j^{k+1}[n]$ respectively, where

$$\begin{aligned}
\hat{\Phi}_j^k[n] &= j\beta_k\omega_o^k(n+\delta^k) + \frac{\Delta_j^k n}{\rho_k} + \phi_j^k \\
\hat{\Phi}_j^{k+1}[n] &= j\beta_{k+1}\omega_o^{k+1}(n+\delta^{k+1}) + \frac{\Delta_j^{k+1} n}{\rho_k} + \phi_j^{k+1},
\end{aligned} \tag{6.21}$$

and denoting the unmodified phase terms from Equation 4.4 as $\Phi_j^k[n]$ and $\Phi_j^{k+1}[n]$, a reasonable constraint on the phase behavior of corresponding components from each synthetic contribution is to require that the differential between the unmodified phase terms at the center of the unmodified synthesis frame match the differential between the modified phase terms at the modified frame center. Formally, this requirement is given by

$$\hat{\Phi}_j^{k+1}[-\rho_k N_s/2] - \hat{\Phi}_j^k[\rho_k N_s/2] = \Phi_j^{k+1}[-N_s/2] - \Phi_j^k[N_s/2], \quad \text{for all } j. \tag{6.22}$$

Solving this equation for δ^{k+1} using the phase functions just defined yields the recursion

$$\delta^{k+1} = \frac{\beta_k \omega_o^k}{\beta_{k+1}\omega_o^{k+1}}(\delta^k + (\rho_k - 1/\beta_k)N_s/2) + (\rho_k - 1/\beta_{k+1})N_s/2. \tag{6.23}$$

Note that there is no dependence on j in this recursion, verifying that δ^{k+1} is a global time shift that needs to be calculated only once per frame. The advantage of this approach to maintaining temporal phase coherence is that it requires much less computation than the coherence preservation algorithm, since no estimate of pitch onset time is required.

CHAPTER 7

Comparative Testing of the ABS/OLA System and the Sine-wave Transform System

At this point the Analysis-by-Synthesis/Overlap-Add system and its associated algorithms have been completely described as applied to the problems of speech modification and music synthesis. However, any discussion of this new system is incomplete without considering its effects in terms of performance relative to competing approaches to the same problems. While this question has been partially answered in Chapter 5 by considering the computational cost of the ABS/OLA system in comparison to the Sine-wave Transform System, a full discussion of performance issues also requires comparison in terms of the quality of processed speech produced by the two systems. This chapter thus compares the ABS/OLA and Sine-wave Transform systems using both objective and formal subjective criteria, and attempts to interpret the results of these tests.

7.1 Objective Testing

As has been noted throughout the thesis, many approaches to speech modeling analyze speech signals by attempting to minimize some error norm in terms of model parameters. This is the case for both analysis-by-synthesis and the peak-picking procedure used in the STS. Analysis-by-synthesis operates by directly minimizing

the segmental squared error norm of Equation 3.16 in terms of the parameters of a synthetic waveform which is a sum of constant-amplitude, constant-frequency sinusoids. This is equivalent to maximizing the segmental SNR defined in Equation 3.34. Similarly, peak-picking uses stationarity arguments to justify choosing spectral peak parameters to minimize the same error norm using the same synthetic waveform [4]. Since both competing techniques attempt to minimize the same segmental SNR measure, comparing their performance in terms of this measure is a fair objective test of modeling accuracy.

Based on this observation, the following test methodology was defined to test different analysis procedures: A given analysis technique was applied to a variety of equal length speech utterances sampled at 8000 samples/sec to determine parameters for the overlap-add sinusoidal model defined in Equation 3.2 with $\sigma[n] \equiv 1$; a value of $N_s = 80$ (10 msec) was used in the overlap-add sinusoidal model, which approximated each utterance with a given fixed number of components. Using the parameters determined by analysis, a synthetic version of each utterance was generated. Given the original and synthetic utterances, segmental SNR was calculated in each synthesis frame, and the average taken over all frames in the utterance served as a measure of accuracy for the analysis technique applied to that utterance. Finally, the average of this measure over all utterances yielded the performance of a given analysis technique for a given number of components.

Three different analysis techniques were evaluated using the procedure described above. They were:

1. *Peak-Picking*, which is described in Section 1.3 and in [4]. The input signal is windowed using a pitch-adaptive Hamming window, zero-padded to 512 samples, then sent to an FFT routine to calculate the 512-point DFT. This high-resolution DFT is then analyzed to determine the frequencies of significant spectral peaks; the peak frequencies and the magnitudes and phases of the DFT at those frequencies are chosen as the model parameters.

2. *Analysis-by-Synthesis.* This is the technique described in Section 3.3, using a pitch-adaptive Hamming window as in peak-picking and the frequency blanking technique described in Section 4.1 with $\gamma_b = .75$ to account for perceptual factors.

3. *ABS-PP* technique. There are two major differences between analysis-by-synthesis and peak-picking: The first is that peak-picking assumes spectral peak frequencies are optimal, while analysis-by-synthesis searches for optimal frequencies. The second is that peak-picking ignores sidelobe interference effects in determining model parameters, while analysis-by-synthesis explicitly accounts for and counteracts those effects. This technique uses the peak frequencies determined by peak-picking as a pruned frequency ensemble, then applies analysis-by-synthesis to determine amplitude and phase parameters; the technique is designed to assess how the differences between peak-picking and analysis-by-synthesis contribute to performance differences.

Figure 7.1 illustrates the average segmental SNR achieved by the three analysis techniques described above as a function of the number of sinusoids used to model speech. As expected, all three techniques display uniformly poor performance using low numbers of components. Furthermore, the SNR of each improves with increasing numbers of sinusoids, and performance gains decrease with each additional component used in the model. Beyond these similarities, though, analysis-by-synthesis displays better SNR levels than peak-picking for all numbers of components tested, with an increase of approximately 5 dB in SNR for more than 30 components.

As interesting, however, is the only marginal improvement in average SNR achieved using peak-picking to determine component frequencies in analysis-by-synthesis. This result suggests that, in terms of mean approximation accuracy, the major drawback of peak-picking analysis is inaccuracy caused by estimating incorrect component frequencies. However, average SNR figures do not tell the entire story, since the ABS-PP technique yields much better approximations of transitory speech events

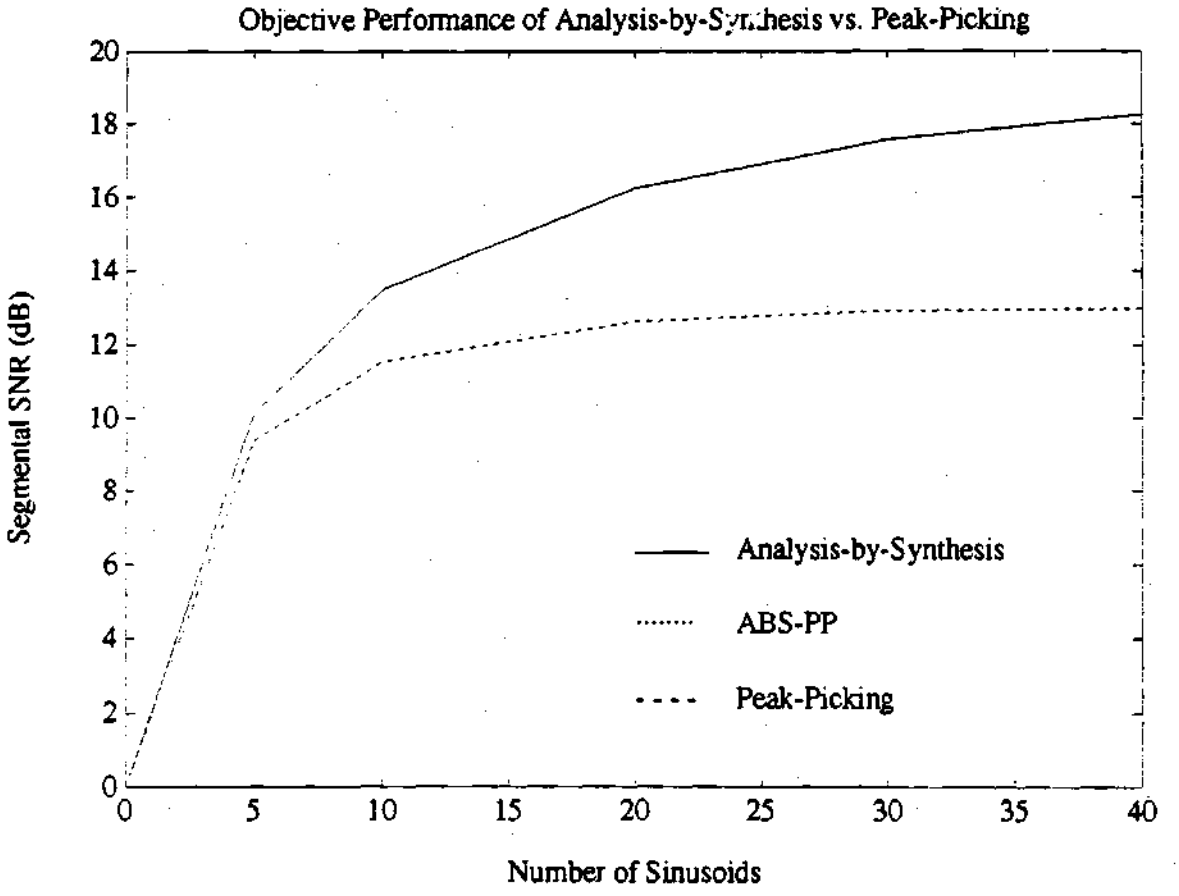

Figure 7.1: Plots of average segmental SNR versus number of sinusoids for three different analysis techniques.

such as plosives than does peak-picking. This is due to sidelobe interference effects for which analysis-by-synthesis can compensate, but which peak-picking ignores. While perceptually important, such speech events are relatively infrequent and thus have little effect on average SNR.

7.2 Subjective Testing

As mentioned in Section 3.3.1, objective measures of accuracy often correlate poorly with subjective listener preferences. An example of this phenomenon is seen in the above paragraph, where infrequent errors in the synthetic speech signal have a disproportionate effect on perceived quality due to their "noticeability." Objective measures tend to have difficulty accounting for such subtleties of human aural perception, casting doubts on their ability to discriminate between the subjective performance of speech processing techniques with comparable scores. To make matters worse, modified speech signals have no reference signal on which to base measurements, rendering signal-based comparison between modification techniques impossible.

Of course, listener preferences are the most important quality measure for speech synthesis and modification systems; therefore, some effort must be made to compare the ABS/OLA and Sine-wave Transform systems on the basis of subjective quality. The approach taken in this work is to use statistical testing to establish mean listener preference scores for various systems over a reasonably broad set of test subjects, and to use hypothesis testing techniques to determine if score differences are statistically meaningful.

A variation of the Paired Acceptability Rating Method (PARM) [85] was used to quantify relative listener preferences between the ABS/OLA system and the Sine-wave Transform System for speech analysis/synthesis and in the presence of various speech modifications. The PARM is an isometric speech quality test in which listeners rate the subjective quality of speech processing systems by assigning scores to sentences

presented in pairs. As with other isometric speech quality measures, PARM testing is most effective when the systems under test exhibit similar types of distortion [86]. For a given type of speech modification, both the ABS/OLA and Sine-wave Transform systems produce similar alterations in perceptual quality, thus PARM results are a useful measure for comparison.

However, comparisons between time-, frequency- or pitch-scale modification systems are not as meaningful, since different types of artifacts result from these speech modifications. Therefore, PARM testing in this work is organized in several PARM modules, with each module corresponding to a single type of modification. A PARM module rates a sentence processed by a set of six different systems in terms of listener preference scores on a 0 to 100 scale. Two of the systems are references designed to normalize testing results and minimize listener bias. A high anchor, the original speech, is assigned a score of 80; a poor-quality low anchor, which in this work was speech approximated with five sinusoids, is assigned a score of 20. The high and low anchors and their scores were presented to listeners periodically during each test.

Listeners were asked to rate processed sentences presented exhaustively in distinct pairs, thus the total number of pairs presented in each test was

$$\binom{6}{2} = 15,$$

and a single system was presented five times in each test. To avoid training the listeners to recognize systems under test, sentences were presented in random order. To minimize bias due to interactions of the tested systems with given sentences or speakers, three unrelated sentences spoken by low pitched male, high pitched male and female speakers were run in independent tests using the same systems. A total of fifteen untrained listeners[1] participated in the testing. Each system under test is thus associated with a total of 225 listener preference scores.

[1] The listeners included graduate students, faculty and staff members in the School of Electrical Engineering at Georgia Tech.

The test results are assumed to follow a normal distribution, and are interpreted in terms of the mean preference score and the standard deviation about the mean. The mean score provides a primary measure of subjective quality, and the standard deviation indicates the amount of agreement among listeners with the mean score.

To be certain that differences between scores for different systems are not due to statistical artifact, the Newman-Keuls test for statistical significance [87] was applied to the results from each PARM module. This test is defined as follows: A collection of scores for the i-th system in a module is denoted as

$$\{S_1^i, \ldots, S_{J_s}^i\},$$

where $J_s = 225$ is the number of sample scores associated with the system. Since the *score distribution* above is assumed to be normal, its statistics are described by the mean and variance

$$\begin{aligned} \overline{S}[i] &= \frac{1}{J_s}\sum_{j=1}^{J_s} S_j^i \\ \sigma^2[i] &= \frac{1}{J_s}\sum_{j=1}^{J_s}\{S_j^i - \overline{S}[i]\}^2; \end{aligned}$$

the score distributions are then rank ordered according to increasing mean. The *unbiased mean-square error* measure for the module is then defined as

$$MS_{\text{err}} \triangleq \frac{J_s}{J_s - 1}\sigma_{\text{av}}^2, \tag{7.1}$$

where the average variance is given by

$$\sigma_{\text{av}}^2 = \frac{1}{I}\sum_{i=1}^{I}\sigma^2[i]$$

and where $I = 6$ is the number of systems under test. The error measure MS_{err} is interpreted as an indication of the consistency of test scores among the systems in a given PARM module.

For a pair of systems $[l, m]$ (where $l > m$) in the PARM module, the following statistic is defined:

$$q_{lm} \triangleq \frac{\overline{S}[l] - \overline{S}[m]}{\sqrt{MS_{\text{err}}/n}}. \tag{7.2}$$

To determine the level of significance between systems l and m, q_{lm} is compared to the tabulated *studentized range statistic* $q_\alpha(N_f, k)$, where $N_f = I(J_s - 1)$ is the number of degrees of freedom in the test and where $k = l - m + 1$. For a given value of α, the range statistic establishes a threshold which q_{lm} must exceed to achieve a confidence percentage of 100α.

Confidence percentages are interpreted as follows: A confidence of 99 percent between two systems implies the probability that random chance could account for different mean scores is less than one percent, which is a widely accepted benchmark of statistical significance. A confidence of 95 percent is considered borderline, while less than 95 percent confidence implies no significant difference between systems. The result of the Newman-Keuls test is a matrix showing percentage of confidence between mean scores of the various systems, with insignificant differences denoted by an asterisk in the matrix.

7.2.1 Interpretation of Test Results

Table 7.1 illustrates the results of PARM Test 1. This test repeats the comparison of analysis-by-synthesis and peak-picking in Section 7.1, this time using a subjective measure. The test compares synthetic speech produced using an overlap-add sinusoidal model with fixed numbers of components, using analysis-by-synthesis or peak-picking to determine model parameters. The systems under test are denoted ABS_N for analysis-by-synthesis with N components and PP_N for peak-picking with N components. Twenty and thirty components were used in the tested systems.

The test results indicate a 5 point difference between system ABS_30 and system PP_30, and the Newman-Keuls test demonstrates that this difference is significant. The difference between analysis-by-synthesis and peak-picking is less pronounced for 20 sinusoids, because [illegible] by too few components masks differences between the analysis [illegible] the standard deviation was relatively small, indicating [illegible].

Another important result from Test 1 is the fact that analysis-by-synthesis with 30 sinusoids scores very nearly the same as original speech. The difference is statistically significant, but by a very narrow margin. Furthermore, referring to Figure 7.1, further gains in accuracy using analysis-by-synthesis can be expected by raising the number of components slightly, while the performance curve of peak-picking is nearly flat at 30 sinusoids.

Tables 7.2–7.4 illustrate comparisons between the fully developed ABS/OLA system (denoted OLA) described in the thesis and the fully developed Sine-wave Transform System (denoted STS) discussed in [4, 5, 43] and in [67]–[40], for isolated time-, frequency- and pitch-scale modifications. Table 7.2 compares the two systems for time-scale modification. The tested systems are denoted OLA_TS_ρ and STS_TS_ρ for time-scale expansion factors $\rho = 2$ and 3.

Table 7.2 shows that the ABS/OLA system achieves a three point gain over the Sine-wave Transform System for a time scale factor of 2, and a four point gain when $\rho = 3$. This indicates that the ABS/OLA system performs better in this application. and that its performance does not break down as rapidly as the STS for large time scale factors. The confidence matrix shows that all results are significant.

Note that the standard deviations of modified speech are all rather high, indicating disagreement among listeners as to quality. This is to be expected, since modification of speech is itself a "distortion" of sorts, and its effect is difficult to separate from modification artifacts in listening tests. Also note that even though there is considerable overlap between score distributions in the test, mean scores differences are nevertheless considered significant. This is due to the fact that the Newman-Keuls test does not require the same separation to discriminate between distributions as might be required of a pattern classifier.

Table 7.3 compares the performance of the ABS/OLA and Sine-wave Transform systems for frequency-scale modification. Systems are denoted OLA_FS_β and

(a)

PARM1 System	Mean Score	Standard Deviation
HIGH ANCHOR	75.71	8.35
ABS_30	73.27	7.76
PP_30	68.67	8.92
ABS_20	66.83	10.51
PP_20	65.29	10.13
LOW ANCHOR	23.02	7.78

(b)

	HIGH	ABS_30	PP_30	ABS_20	PP_20
ABS_30	99				
PP_30	99	99			
ABS_20	99	99	95		
PP_20	99	99	99	*	
LOW ANCHOR	99	99	99	99	99

Table 7.1: Results of PARM Test 1, testing analysis-by-synthesis against peak-picking: *(a)* Statistics of system scores in Test 1, listed in order of decreasing mean; *(b)* Matrix showing level of confidence between systems.

(a)

PARM2 System	Mean Score	Standard Deviation
HIGH ANCHOR	78.04	4.28
OLA_TS_2	52.84	13.49
STS_TS_2	49.72	13.92
OLA_TS_3	44.99	13.87
STS_TS_3	40.88	15.08
LOW ANCHOR	25.09	10.10

(b)

	HIGH	OLA_TS_2	STS_TS_2	OLA_TS_3	STS_TS_3
OLA_TS_2	99				
STS_TS_2	99	99			
OLA_TS_3	99	99	99		
STS_TS_3	99	99	99	99	
LOW ANCHOR	99	99	99	99	99

Table 7.2: Results of PARM Test 2, comparing ABS/OLA and STS for time-scale modification: *(a)* Statistics of system scores in Test 2, listed in order of decreasing mean; *(b)* Matrix showing level of confidence between systems.

STS_FS_β for frequency-scale factors[2] $\beta = .75$ and 1.33. The results show an insignificant difference between the systems when $\beta = .75$, due to compression of information into the low frequency band and the resulting masking of modification artifacts. However, the ABS/OLA system gains four points over the Sine-wave Transform System when $\beta = 1.33$, at which point the shortcomings of the STS become more apparent.

Table 7.4 compares the performance of the ABS/OLA and Sine-wave Transform systems for pitch-scale modification. Systems are denoted OLA_PS_β and STS_PS_β for pitch-scale factors $\beta = .75$ and 1.33. Table 7.4 shows a dramatic performance improvement for the ABS/OLA system of 25 points when pitch is raised and 15 points when the pitch is lowered. Perhaps the most significant result of Table 7.4 is the scores of ABS/OLA pitch modification, which are the highest of all modification systems and not vastly different from the high anchor score. This suggests that the ABS/OLA system is largely successful in capturing and preserving speech quality, naturalness and intelligibility in the presence of pitch modification.

However, the low perfomance figures of the STS for pitch-scale modification come with a serious caveat: Pitch-scale modification is the least documented feature of the STS, hence it is likely that significant undocumented improvements have been built into more recent versions of the system for this application. Indeed, while the ABS/OLA system demonstrates clear performance improvements over the STS for the applications of speech analysis/synthesis and speech modification, it is worth mentioning that the version of the Sine-wave Transform System tested here was implemented from published results and from private communications with Drs. McAulay and Quatieri; since certain implementation details are unpublished and often proprietary, it is possible that comparison of the ABS/OLA System with the Sine-wave Transform System as implemented at Lincoln Laboratories would be more competitive than shown here.

[2]These values correspond to lowering and raising pitch 5 half steps on the musical scale.

(a)

PARM3 System	Mean Score	Standard Deviation
HIGH ANCHOR	75.85	7.22
OLA_FS_.75	58.66	14.57
STS_FS_.75	57.46	14.38
OLA_FS_1.33	56.38	12.28
STS_FS_1.33	52.01	12.35
LOW ANCHOR	24.33	8.23

(b)

	HIGH	OLA_FS_.75	STS_FS_.75	OLA_FS_1.33	STS_FS_1.33
OLA_FS_.75	99				
STS_FS_.75	99	*			
OLA_FS_1.33	99	*	*		
STS_FS_1.33	99	99	99	99	
LOW ANCHOR	99	99	99	99	99

Table 7.3: Results of PARM Test 3, comparing frequency-scale modification: *(a)* Statistics of system scores in Test 3, listed in order of decreasing mean; *(b)* Matrix showing level of confidence between systems.

(a)

PARM4 System	Mean Score	Standard Deviation
HIGH ANCHOR	73.25	9.91
OLA_PS_1.33	62.56	11.36
OLA_PS_.75	61.25	11.92
STS_PS_.75	46.02	16.27
STS_PS_1.33	37.55	13.44
LOW ANCHOR	24.99	9.18

(b)

	HIGH	OLA_PS_1.33	STS_PS_1.33	OLA_PS_.75	STS_PS_.75
OLA_PS_1.33	99				
STS_PS_1.33	99	99			
OLA_PS_.75	99	*	99		
STS_PS_.75	99	99	99	99	
LOW ANCHOR	99	99	99	99	99

Table 7.4: Results of PARM Test 4, comparing pitch-scale modification: *(a)* Statistics of system scores in Test 4, listed in order of decreasing mean; *(b)* Matrix showing level of confidence between systems.

CHAPTER 8

Conclusions

This chapter concludes the discussion of Analysis-by-Synthesis/Overlap-Add sinusoidal modeling by reviewing the work completed to date, interpreting the results of research presented in the thesis, and suggesting possible areas of future research.

8.1 Review of Major Results

The major objectives of the research presented in this thesis were: (1) to explore the use of analysis-by-synthesis in conjunction with an overlap-add sinusoidal model for the applications of speech analysis/synthesis, speech modification and digital music synthesis; (2) to develop working speech modification and music synthesis systems based on the results of this research; and (3) to evaluate the effectiveness of the systems developed by way of comparison with similar results generated using the Sine-wave Transform System of McAulay and Quatieri.

The work of this thesis attempted to answer several key questions concerned with these objectives: First, could analysis-by-synthesis be formulated in the context of sinusoidal modeling in a useful and meaningful way? Second, could overlap-add sinusoidal modeling be formulated to be used effectively for the applications of speech modification and music synthesis? Third, could a system based on analysis-by-synthesis overlap-add sinusoidal modeling be implemented in a computationally efficient manner? Finally, could the use of this fully developed modification system be supported by improvements in the subjective quality of synthetic and modified

speech relative to the quality produced using the Sine-wave Transform System and by favorable computational comparisons with this popular modification system?

To meet these objectives and answer these questions, the thesis research began with an investigation of iterative vector approximation, an approach to vector-space approximation which forms the mathematical basis of many analysis-by-synthesis techniques. This discussion served to cast the implementation of analysis-by-synthesis in a general framework, and allowed the use of results from vector space theory and linear algebra to analyze the performance of analysis-by-synthesis.

For instance, a simple condition was derived which guarantees approximation convergence in analysis-by-synthesis, and an equally simple mutual orthogonality condition was determined for which analysis-by-synthesis produces an optimal approximation. In addition, it was found that the effectiveness of iterative vector approximation could be evaluated directly in terms of the amount of cross-correlation of vectors used in the approximation.

The research continued with the definition of an overlap-add sinusoidal model for audio signals similar to other overlap-add model formulations, but with the addition of a modulating envelope sequence to improve model performance in signals which exhibit temporal energy variations. After considering other methods for calculating this envelope sequence, it was found that quasi-Gaussian lowpass filtering resulted in an envelope sequence which tracked energy variations very well while exhibiting very little ripple.

It was then observed that the problem of determining sinusoidal model parameters on a frame-by-frame basis could be cast as a nonlinear least-squares approximation problem. Further study revealed that this problem could easily be cast into the mathematical framework of analysis-by-synthesis discussed above, and on this basis computational methods were derived to implement an analysis-by-synthesis algorithm to determine overlap-add sinusoidal model parameters in a successive manner. In addition, it was determined that this algorithm converges for all but degenerate cases.

Several types of stopping conditions for the analysis algorithm were explored, and a μ-law derived SNR threshold was employed for audio signal analysis.

Since sinusoidal sequences are used in the overlap-add model and associated analysis algorithm, it was found that the analysis-by-synthesis procedure for this model could be expressed in terms of a frequency-domain dual, based on matching spectral values at component frequencies and successive subtraction of circularly-shifted spectra. It was then observed that this frequency-domain interpretation could be used in conjunction with results from iterative vector approximation to establish criteria for choosing an analysis window to minimize cross-correlation and hence maximize the performance of analysis-by-synthesis.

The basic definition of analysis-by-synthesis for sinusoidal modeling employs a segmental signal-to-noise error criterion for approximation; as a result, the subjective quality of analyzed speech signals is often lower than might be expected for a given number of component sinusoids. This effect was quantified in the frequency domain as "clustering" of components with small amplitudes around major components. Several methods of counteracting clustering were explored, including adaptive prefiltering based on the Atal/Schroeder weighting filter [62], fixed prefiltering to account for spectral tilt, and a novel "frequency blanking" algorithm to discourage spurious component frequencies; this latter algorithm was implemented in analysis-by-synthesis due to its simplicity, generality and effectiveness.

Given the fully developed approach to speech analysis, a quasi-harmonic formulation of the overlap-add sinusoidal model was defined to facilitate application of the ABS/OLA system to the problem of speech modification. Based on model parameters determined in analysis-by-synthesis, a new algorithm was defined which simultaneously estimated an optimal fundamental frequency parameter, resolved pitch ambiguities, and organized an appropriate subset of analyzed parameters in quasi-harmonic form. This algorithm has been demonstrated to provide accurate, robust fundamental frequency estimates at a relatively low computational load, assuming

wideband interference.

A considerable amount of research effort went into formulating an overlap-add sinusoidal model which could be used directly to perform speech modifications. The basic difficulty addressed was that of preserving the phase coherence of modified synthetic segments used to construct a modified speech signal. After considering the complex form of sums of sinusoids, it was realized that manipulating component frequencies in the quasi-harmonic model made it possible to explicitly control and guarantee phase coherence in the presence of time- and frequency-scale modifications using very simple frequency relations.

A variation of the pitch onset time excitation model defined by McAulay and Quatieri [39] was used in conjunction with this refined modification model to derive an algorithm to preserve temporal phase coherence in modified speech, based on inter-frame coherence constraints. This algorithm has the significant advantage of being "self-correcting" in the presence of common fundamental frequency estimation errors, greatly improving the ABS/OLA System's robustness to such errors.

Having addressed the basic issues of performing speech modifications using an overlap-add synthesis model, the problem of performing pitch-scale modification using the resulting system was studied next. This investigation yielded an algorithm for pitch modification using the pitch onset time excitation model described above and phasor interpolation of excitation parameters to produce a modified excitation sequence with altered fundamental frequency. By producing excitation parameters across the available spectrum and localizing noise influences, the phasor interpolation algorithm effectively preserves speech bandwidth and avoids noise amplification, two common problems in other approaches to pitch modification.

While the ABS/OLA System is capable of producing high-quality synthetic and modified speech signals, when implemented in direct form it is also extremely computationally complex. Since, as mentioned above, analysis-by-synthesis may be viewed as a frequency-domain approximation problem, it was decided to explore this formu-

lation in an attempt to gain computational advantages. It was soon discovered that frequency-domain analysis-by-synthesis using uniformly spaced candidate frequencies may be expressed as a manipulation of two discrete Fourier transform sequences per analysis frame; since these sequences can be computed using the FFT algorithm, the resulting fast analysis-by-synthesis algorithm requires significantly less computation to implement than the direct approach.

In addition, since synthetic contributions in the overlap-add sinusoidal model are generated from constant-amplitude, constant frequency sinusoids, it was possible to use the IFFT algorithm to perform synthesis. These computational enhancements of the ABS/OLA system make its implementation possible in a near real-time environment with current hardware. The ABS/OLA System was demonstrated to have better computational performance in synthesis than the Sine-wave Transform System (using half-rate overlap-add synthesis), but higher computational load in analysis than the simpler peak-picking algorithm.

After gaining familiarity with the current state of the art in digital music synthesis techniques, a considerable technological tradeoff was noted between simple but crude music synthesis techniques and sophisticated but computationally intense analysis/synthesis algorithms, with few techniques exhibiting reasonable compromises between these extremes. Noting the similarity of the ABS/OLA system to the classical additive synthesis model, it was expected that overlap-add sinusoidal modeling would serve as an appropriate music synthesis technique. Furthermore, this analysis/synthesis system combines high-quality synthesis with a particularly fast synthesis algorithm, an attractive combination for music synthesis applications. Also, using *a priori* pitch information in analysis-by-synthesis, it was possible to optimize analysis performance and to reduce the overhead computation of fundamental frequency estimation and inter-frame coherence constraints.

In order to gauge the effectiveness of the research presented and algorithms developed in this thesis, the last topic addressed was a comparison of the quality of

synthetic and modified speech produced by the ABS/OLA and Sine-wave Transform Systems in a comparative testing environment. An objective comparison between analysis-by-synthesis and peak-picking was performed on the basis of a segmental SNR measure, using parameters determined by the two analysis algorithms; this test indicated a gain of 5-6 dB using analysis-by-synthesis versus peak picking for nominal numbers of components.

In an attempt to quantify the relative subjective performance of the ABS/OLA and Sine-wave Transform systems, formal subjective testing was performed using a variation of the Paired Acceptability Rating Method (PARM). Synthetic speech, as well as time-, frequency- and pitch-scale modified speech produced by both systems was compared in a series of PARM modules: test results were analyzed in terms of their statistical distributions, with the Newman-Keuls test [87] applied to determine statistical significance. These subjective tests demonstrated a marked improvement in the quality of synthetic and modified speech using the ABS/OLA system, with almost all comparative results established as statistically significant.

8.2 Interpretation of Results

The Analysis-by-Synthesis/Overlap-Add system presented in this thesis possesses a number of positive features when applied to the problems of speech analysis/synthesis and speech modification. Since the development of the ABS/OLA system often parallels that of the Sine-wave Transform System, it is particularly useful to highlight these features by way of comparison between the two techniques.

Beginning with synthesis model definitions, consider the overlap-add sinusoidal model used in the ABS/OLA system (Equation 3.2), and the overlap-add model used to derive peak-picking analysis in the STS (Equation 1.10). The difference between these models is the incorporation of an envelope sequence $\sigma[n]$ in the ABS/OLA model to account for [illegible] energy variations; the advantage of using this envelope

is that analysis-by-synthesis is able to explicitly use $\sigma[n]$ to increase performance when signal energy changes rapidly, whereas peak-picking assumes stationarity for all analyzed signal segments. Thus analysis-by-synthesis is capable of more uniform signal approximation than peak-picking.

As mentioned before, a primary difference between the peak-picking and analysis-by-synthesis algorithms is that peak-picking assumes spectral magnitude peaks correspond to optimal frequencies, and that spectral values at these frequencies yield optimal amplitude and phase parameters; by contrast, analysis-by-synthesis attempts to explicitly determine parameters for each component in the approximation which minimize approximation error. There are two main advantages gained by the latter approach: First, since analysis-by-synthesis incorporates an explicit parameter search, it comes closer to optimal performance for a given number of components than peak-picking, particularly when stationarity assumptions break down. Interestingly, as discussed in Section 7.1, the greatest source of approximation error in peak-picking can be traced to errors in frequency estimation; this makes sense when considering how sensitive sinusoidal modeling is to frequency errors.

The second important advantage of analysis-by-synthesis is seen by referring to the frequency-domain analysis-by-synthesis example of Figure 3.6. The top left plot shows a signal spectrum and the "single component" spectrum which best approximates it; the sidelobes of the window spectrum are apparent in the single component spectrum. When a number of components are added together, sidelobes from component spectra will interfere with each other, even if the mainlobes do not overlap, causing slight parameter estimation errors in peak-picking.

While interference is arguably negligible for high-amplitude, low-frequency components, the errors caused by low-frequency component sidelobes on relatively low-amplitude high-frequency components is not so slight; this interference accounts for the "tonal quality" often perceived at high frequencies in the Sine-wave Transform System. The advantage of analysis-by-synthesis is that since component spectra are

successively subtracted from the original spectrum, sidelobe interference is subtracted as well. Thus, this technique has the built-in ability to counteract interference, which eliminates tonal artifacts.

One of the most significant features of the ABS/OLA system is the ability to perform artifact-free speech modification using an overlap-add sinusoidal model. By manipulating quasi-harmonic component frequencies, the refined modification model of the ABS/OLA system manages to preserve phase coherence and waveform shape in the same manner as shape-invariant modification in the STS (as discussed in Section 1.3.1). The main difference between the two approaches is that while the STS requires an interpolated parameter model, the refined modification model maintains exactly the same overlap-add structure used to define the analysis-by-synthesis technique. This correlation between analysis and synthesis techniques in the ABS/OLA system leads to more consistent and predictable quality in speech synthesis and modification.

Another important distinction between the refined modification model of the ABS/OLA system and shape-invariant modification is seen by considering the strategy used in the latter approach: In shape-invariant modification, temporal phase coherence is preserved by introducing a time shift to $\tilde{s}^k[n]$. However, as discussed in Section 4.3.1 and illustrated in Figure 4.5, since component frequencies are not altered to preserve intra-frame coherence, the time shift can cause a breakdown of phase coherence in the STS, particularly for non-stationary speech segments. By contrast, although a similar time-shifting strategy is used in the refined modification model, Equation 4.15 reveals that the time shift is only applied to harmonic components, hence intra-frame phase coherence is unaffected by the shift.

The phasor interpolation algorithm represents a novel approach to performing pitch-scale modification using sinusoidal model parameters and a minimum-phase spectral envelope estimate $H(e^{j\omega})$. Phasor interpolation bears a resemblance to the mixed-phase deconvolution algorithm of the Sine-wave Transform System, in that

residual phase parameters from the pitch onset time excitation model are incorporated in the parameterization of speech, and accounted for in the presence of modification. However, while pitch modification using mixed-phase deconvolution performs frequency-scale modification of the excitation signal, phasor interpolation instead defines an underlying function of frequency based on excitation parameters which is resampled at the new fundamental frequency. As mentioned earlier, this implies both that speech bandwidth is preserved and that noise effects do not migrate (assuming bandlimited interpolation).

As a final point, note that the refined overlap-add model in the ABS/OLA system allows synthesis to be performed using two synthetic contributions per synthesis frame, regardless of the modification performed. By contrast, the Sine-wave Transform System uses a third "mid-frame" synthetic contribution to perform overlap-add synthesis for frame lengths greater than 10 msec. As discussed in Chapter 5, this extra contribution per frame implies that ABS/OLA synthesis is typically faster than STS synthesis.

8.3 Suggestions for Future Research

While the research presented in this thesis has been largely successful in meeting the stated objectives and dealing with the necessary questions involved in meeting these objectives, and while the ABS/OLA system has proven to be useful for the applications of speech analysis/synthesis, speech modification and music synthesis, a number of unresolved issues of performance, computation and application remain to be addressed. This section will summarize these issues and suggest some possible approaches to dealing with them.

The envelope sequence $\sigma[n]$ used in the ABS/OLA system is helpful in achieving accurate approximations of signals, particularly when using the quasi-harmonic model. However, the quasi-Gaussian filtering technique used to determine $\sigma[n]$, while

effective, is computationally expensive. A more efficient filter structure which achieves similar performance would be in order for real-time implementations; an alternative approach would be to determine conditions under which a certain amount of ripple in $\sigma[n]$ could be incorporated into the model without causing distortion.

The computational load of analysis-by-synthesis, while manageable, remains an obstacle for real-time implementation on certain platforms. As pointed out in Chapter 5, part of the computational load in analysis-by-synthesis derives from the need to calculate optimal component parameters for each candidate frequency. As mentioned there, Equation 3.53 suggests that simply choosing the maximum of each successive error spectrum provides a reasonable approximation of the optimal component frequency for most frequency values. However, this approach can result in artifacts due to underapproximation in the low-frequency range.

Most of the computational load in analysis-by-synthesis derives from exhaustive frequency searching. While exhaustive searching tends to optimize analysis accuracy, there is sufficient information in speech and music signal spectra to allow pruning of the search space with little loss of quality. One strategy that will be explored further is to pick a certain number of peaks from the original spectrum, perform analysis-by-synthesis in narrow bands around those frequencies, then repeat the process after removing the analyzed components.

Stopping conditions in analysis-by-synthesis are defined in terms of how close the approximation error is to zero. While this is useful for analysis of signals with little noise interference, significant additive noise can result in overapproximation and the undesirable tendency to capture noise as well as signal. For enhancement applications, it would be helpful to define a different stopping condition based on, for instance, the "whiteness" of the error for the condition of additive white noise.

While the frequency blanking algorithm for perceptual enhancement of analysis-by-synthesis works well for many cases, it results in poor quality speech when $w_a[n]$ is too short. Indeed, the question of perceptual factors in analysis-by-synthesis is an

open question. One possible solution is seen by considering Figure 3.6. As noted in Section 4.1, analysis-by-synthesis using the least-square error norm only accounts for the spectral magnitude at a given candidate frequency; if an error norm were defined which accounts for the match of spectral shape in the mainlobe, then very few spurious components would be chosen, without requiring a hard decision as in frequency blanking.

The phasor interpolation algorithm for pitch-scale modification has several advantages as noted before; however, speech modified using this approach can sound strange for significant alterations of pitch. To deal with this problem, the pitch modification algorithm requires refinement to account for vocal tract changes which occur in human speech production at different pitch frequencies [25]. A more significant problem is the sensitivity of phasor interpolation to pitch estimation errors and the artifacts which often result. To deal with this problem, the causes of this sensitivity should be isolated and eliminated.

Finally, it is worth noting that while speech modification and music synthesis were the primary application areas developed in this thesis, they are by no means the only area to which the ABS/OLA system may be applied. Currently, several other applications, such as helium speech enhancement, co-channel speaker separation, and speaker normalization for speech recognition are being explored. Also, the application of low bit-rate speech coding is worthy of special note. While sinusoidal modeling was originally developed for this application, the problem of parameterizing and coding phase has always been a stumbling block; the discussion of phase coherence issues for speech modification using an overlap-add sinusoidal model suggests a possible starting point for a low-dimensional parameterization of phase information.

Bibliography

[1] J. D. Markel and A. H. Gray Jr. *Linear Prediction of Speech.* Springer-Verlag, New York, New York, 1976.

[2] J. A. Moorer. The Use of the Phase Vocoder in Computer Music Applications. *J. Audio Eng. Soc.*, 26(1/2):42–45, January/February 1978.

[3] M. Dolson. The Phase Vocoder: A Tutorial. *Computer Music Journal*, 10(4):14–27, Winter 1986.

[4] R. J. McAulay and T. F. Quatieri. Speech Analysis/Synthesis Based on a Sinusoidal Representation. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP–34(4):744–754, August 1986.

[5] T. F. Quatieri and R. J. McAulay. Speech Transformations Based on a Sinusoidal Representation. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP–34(6):1449–1464, December 1986.

[6] R. J. McAulay and T. F. Quatieri. Multirate Sinusoidal Transform Coding at Rates From 2.4 Kbps to 8Kbps. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 1645–1648, April 1987.

[7] T. F. Quatieri and R. G. Danisewicz. An Approach to Co-Channel Talker Interference Suppression Using a Sinusoidal Model for Speech. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 565–568, April 1988.

[8] T. F. Quatieri and R. J. McAulay. Noise Reduction Using a Soft-Decision Sine-Wave Vector Quantizer. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 821–824, April 1990.

[9] B. S. Atal and J. R. Remde. A New Model of LPC Excitation for Producing Natural-Sounding Speech at Low Bit Rates. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 614–617, May 1982.

[10] B. S. Atal. Predictive Coding of Speech at Low Bit Rates. *IEEE Trans. on Comm.*, COM-30(4):600–614, April 1982.

[11] M. R. Schroeder and B. S. Atal. Code-Excited Linear Prediction (CELP): High-Quality Speech at Very Low Bit Rates. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 937–940, April 1985.

[12] R. C. Rose and T. P. Barnwell III. The Self-Excited Vocoder – An Alternate Approach to Toll Quality at 4800 bps. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 453–456, April 1986.

[13] J. Makhoul. Linear Prediction: A Tutorial Review. *Proc. IEEE*, 63(4):561–580, April 1975.

[14] C. J. Weinstein. A linear prediction vocoder with voice excitation. In *Proc. EASCON*, pages 30A–30G, September 1975.

[15] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*, page 398. Prentice-Hall, Englewood Cliffs, New Jersey, 1978.

[16] D. H. Klatt. Synthesis-by-Rule of Segmental Durations in English Sentences. In *Proc. 9th Int'l Congress of Phonetic Sciences*, Copenhagen, 1979.

[17] J. L. Flanagan. Spectrum analysis in speech coding. *IEEE Trans. Audio Electroacoust.*, AU-15:66–69, June 1967.

[18] J. L. Flanagan. *Speech Analysis, Synthesis and Perception.* Springer-Verlag, New York, New York, 1972.

[19] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*, pages 250–276. Prentice-Hall, Englewood Cliffs, New Jersey, 1978.

[20] R. W. Schafer and L. R. Rabiner. Design of Digital Filter Banks for Speech Analysis. *Bell Sys. Tech. J.*, 50(10):3097–3115, December 1971.

[21] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*, pages 718–721. Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[22] J. L. Flanagan and R. M. Golden. Phase Vocoder. *Bell Sys. Tech. J.*, 45:1493–1509. 1966.

[23] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*, pages 334–341. Prentice-Hall, Englewood Cliffs, New Jersey, 1978.

[24] L. D. Braida et al. Matching Speech to Residual Auditory Function - A Review of Past Research. ASHA Monograph, 1978.

[25] S. Seneff. System to Independently Modify Excitation and/or Spectrum of Speech Waveform Without Explicit Pitch Extraction. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-30(4):566–578, August 1982.

[26] B. P. Boger et al. The Quefrency Alanysis of Time Series For Echoes: Cepstrum, Pseudo-autocovariance, Cross-Cepstrum, and Saphe Cracking. In M. Rosenblatt, editor, *Proc. Symp. Time Series Analysis*, pages 209–243. John Wiley and Sons, Inc., New York, New York, 1963.

[27] D. Malah. Time-Domain Algorithms for Harmonic Bandwidth Reduction and Time Scaling of Speech Signals. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-27(2):121–133, April 1979.

[28] R. W. Schafer and L. R. Rabiner. Design and Simulation of a Speech Analysis-Synthesis System Based on Short-Time Fourier Analysis. *IEEE Trans. Audio Electroacoust.*, AU-21(3):165–174, June 1973.

[29] J. B. Allen. Short Term Spectral Analysis, Synthesis and Modification by Discrete Fourier Transform. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-25(3):235–238, June 1977.

[30] R. E. Crochiere. A Weighted Overlap-Add Method of Short-Time Fourier Analysis/Synthesis. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-28:99–102, February 1980.

[31] M. R. Portnoff. Implementation of the Digital Phase Vocoder Using the Fast Fourier Transform. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-24(3):243–248, June 1976.

[32] M. R. Portnoff. *Time-Scale Modification of Speech Based on Short-Time Fourier Analysis.* PhD thesis, Massachusetts Institute of Technology, 1978.

[33] D. W. Griffin and J. S. Lim. Signal Estimation from Modified Short-Time Fourier Transform. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-32(2):236–242, 1984.

[34] M. A. Jasiuk et al. Improved Speech Modification Method. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 1465–1468, April 1987.

[35] S. Roucous and A. M. Wilgus. High-Quality Time-Scale Modification of Speech. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 493–496, April 1985.

[36] P. Hedelin. A Tone-Oriented Voice-Excited Vocoder. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 205–208, March 1981.

[37] L. B. Almeida and F. M. Silva. Variable-Frequency Synthesis: An Improved Harmonic Coding Scheme. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, page 27.5.1, April 1984.

[38] A. H. Nuttall. Some Windows with Very Good Sidelobe Behavior. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-29(1):84–91, February 1981.

[39] R. J. McAulay and T. F. Quatieri. Phase Modelling and its Application to Sinusoidal Transform Coding. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 1713–1715, April 1986.

[40] T. F. Quatieri and R. J. McAulay. Phase Coherence in Speech Reconstruction for Enhancement and Coding Applications. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 207–210, May 1989.

[41] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*, pages 240–250. Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[42] T. F. Quatieri and R. J. McAulay. Mixed-Phase Deconvolution of Speech Based on a Sine-Wave Model. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 649–652, April 1987.

[43] R. J. McAulay and T. F. Quatieri. Computationally Efficient Sine-Wave Synthesis and its Application to Sinusoidal Transform Coding. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 370–373, April 1988.

[44] J. M. Chowning. The Synthesis of Complex Audio Spectra by Means of Frequency Modulation. *J. Audio Eng. Soc.*, 21(7):526–534, September 1973.

[45] J. A. Moorer. Signal Processing Aspects of Computer Music: A Survey. *Proc. IEEE*. 65(8):1108–1137, August 1977.

[46] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*, pages 407–411. Prentice-Hall, Englewood Cliffs, New Jersey, 1978.

[47] B. Noble and J. W. Daniel. *Applied Linear Algebra*, page 134. Prentice-Hall, Englewood Cliffs, New Jersey, second edition, 1977.

[48] G. H. Golub and C. F. Van Loan. *Matrix Computations*, pages 584–585. Johns Hopkins University Press, Baltimore, Maryland, second edition, 1989.

[49] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*, pages 120–126. Prentice-Hall, Englewood Cliffs, New Jersey, 1978.

[50] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*, pages 194–200. McGraw-Hill, New York, New York, second edition, 1984.

[51] S. M. Kay and S. L. Marple. Spectrum Analysis–A Modern Perspective. *Proc. IEEE*, 69(11):1380–1419, November 1981.

[52] D. W. Tufts and R. Kumaresan. Estimation of Frequencies of Multiple Sinusoids: Making Linear Prediction Perform Like Maximum Likelihood. *Proc. IEEE*, 70(9):975–989. September 1982.

[53] C. G. Bell et al. Reduction of Speech Spectra by Analysis-by-Synthesis Techniques. *J. Acoust. Soc. Am.*, 33:1725–1736, December 1961.

[54] E. B. George and M. J. T. Smith. A New Speech Coding Model Based on a Least-Squares Sinusoidal Representation. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 1641–1644, April 1987.

[55] E. B. George and M. J. T. Smith. Perceptual Considerations in a Low Bit Rate Sinusoidal Vocoder. In *Proc. IEEE Int'l Phoenix Conf. on Computers in Communications*, pages 268–275, March 1990.

[56] E. B. George and M. J. T. Smith. An Analysis-by-Synthesis Approach to Sinusoidal Modeling Applied to the Analysis and Synthesis of Musical Tones. Manuscript submitted to *J. Audio Eng. Soc.*, December 1990.

[57] S. R. Quackenbush, T. P. Barnwell III, and M. A. Clements. *Objective Measures of Speech Quality*. Prentice-Hall, 1988.

[58] B. Smith. Instantaneous Companding of Quantized Signals. *Bell Sys. Tech. J.*, 36(3):653–709, May 1957.

[59] A. V. Oppenheim et al. *Signals and Systems*, pages 333–335. Prentice-Hall, Englewood Cliffs, New Jersey, 1983.

[60] J. F. Kaiser. Nonrecursive digital filter design using the I_0-sinh window function. In *Proc. IEEE Int'l Symp. on Circuits and Systems*, pages 20–23, April 1974.

[61] J. D. Durrant and J. H. Lovrinic. *Bases of Hearing Science*, page 240. Williams and Wilkins, Baltimore, Maryland, 1984.

[62] B. S. Atal and M. R. Schroeder. Predictive Coding of Speech and Subjective Error Criteria. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-27(3):247–254, June 1979.

[63] J. M. Pickett. *The Sounds of Speech Communication*, pages 60–65. University Park Press, Baltimore, Maryland, 1980.

[64] J. C. Rutledge. *Time-Varying, Frequency-Dependent Compensation for Recruitment of Loudness*. PhD thesis, Georgia Institute of Technology, 1989.

[65] Y. Medan et al. Super Resolution Pitch Determination of Speech Signals. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-39(1):40–48, January 1991.

[66] J. D. Markel. The SIFT Algorithm for Fundamental Frequency Estimation. *IEEE Trans. on Audio and Electroacoustics*, AU-20(5):367–377, December 1972.

[67] R. J. McAulay and T. F. Quatieri. Pitch Estimation and Voicing Detection Based on a Sinusoidal Speech Model. In *Proc. IEEE Int'l Conf. on Acoust., Speech and Signal Processing*, pages 249–252, April 1990.

[68] D. W. Griffin and J. S. Lim. Multiband Excitation Vocoder. *IEEE Trans. on Acoust., Speech and Signal Processing*, ASSP-36(8):1223-1235, August 1988.

[69] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*, pages 527-530. Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[70] M. V. Mathews et al. *The Technology of Computer Music*, pages 49-62. MIT Press, Cambridge, Massachusetts, 1969.

[71] J.-C. Risset and M. V. Mathews. Analysis of Musical Instrument Tones. *Physics Today*. 22(2):23-30, 1969.

[72] W. Kaegi and S. Tempelaars. VOSIM - A New Sound Synthesis System. *J. Audio Eng. Soc.*, 26(6):418-425, June 1978.

[73] X. Rodet. Time Domain Formant Wave-Function Synthesis. In J. C. Simon, editor, *Spoken Language Generation and Understanding*. D. Reidel, Dordrecht, Holland, 1980.

[74] G. Bennett and X. Rodet. Synthesis of the Singing Voice. In M. V. Mathews and J. R. Pierce, editors, *Current Directions in Computer Music Research*. MIT Press, Cambridge, Massachusetts, 1989.

[75] B. Truax. Organizational Techniques for C:M Ratios in Frequency Modulation. *Computer Music Journal*, 1(4):39-45, Winter 1977.

[76] L. W. Couch II. *Digital and Analog Communication Systems*, pages 193-195. Macmillan, New York, New York, 1983.

[77] M. Le Brun. Digital Waveshaping Synthesis. *J. Audio Eng. Soc.*, 27(4):250-266, April 1979.

[78] D. Arfib. Digital Synthesis of Complex Spectra by Means of Multiplication of Nonlinear Distorted Sine Waves. *J. Audio Eng. Soc.*, 27(10):757-768, October 1979.

[79] J. A. Moorer. The Synthesis of Complex Audio Spectra by Means of Discrete Summation Formulas. *J. Audio Eng. Soc.*, 24(9):717–727, November 1976.

[80] M. E. McIntyre and J. Woodhouse. On the Fundamentals of Bowed String Dynamics. *Acustica*, 43(2):93–108, September 1979.

[81] M. E. McIntyre, R. T. Schumacher, and J. Woodhouse. On the Oscillations of Musical Instruments. *J. Acoust. Soc. Am.*, 74(5):1325–1345, 1983.

[82] K. Karplus and A. Strong. Digital Synthesis of Plucked-String and Drum Timbres. *Computer Music Journal*, 7(2):43–55, Summer 1983.

[83] L. Hiller and P. Ruiz. Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects (Part 1). *J. Audio Eng. Soc.*, 19(6):462–470, June 1971.

[84] D. A. Jaffe and J. O. Smith. Extensions of the Karplus-Strong Plucked-String Algorithm. *Computer Music Journal*, 7(2):56–69, Summer 1983.

[85] W. D. Voiers. Methods of Predicting User Acceptance of Voice Communications Systems. Final Rep. 100-74-C-0056, DCA, April 1976.

[86] R. C. Rose. *The Design and Performance of An Effective Class of Predictive Speech Coders.* PhD thesis, Georgia Institute of Technology, 1988.

[87] B. J. Winer. *Statistical Principles in Experimental Design.* McGraw-Hill, New York, New York, 1962.

Vita

E. Bryan George was born on December 30, 1960 in Durham, North Carolina. He attended public schools in Milledgeville, Georgia, where he graduated from Baldwin County High School in 1979.

Mr. George received the Bachelor of Electrical Engineering degree from the Georgia Institute of Technology in June, 1985. During his undergraduate studies he was employed as a co-operative student at Georgia Power Company. In June 1985, he enrolled in the graduate program at Georgia Tech, where he received his Ph.D. in Electrical Engineering in December, 1991. He is currently employed with the Signal Processing Center of Technology, Lockheed Sanders Inc., Nashua, NH.

His current research interests include speech processing, general digital signal processing, applications of signal processing to communications, and musical applications of signal processing. He is a member of the Institute of Electrical and Electronic Engineers' Signal Processing and Communication Societies, the Computer Music Association, and Tau Beta Pi.